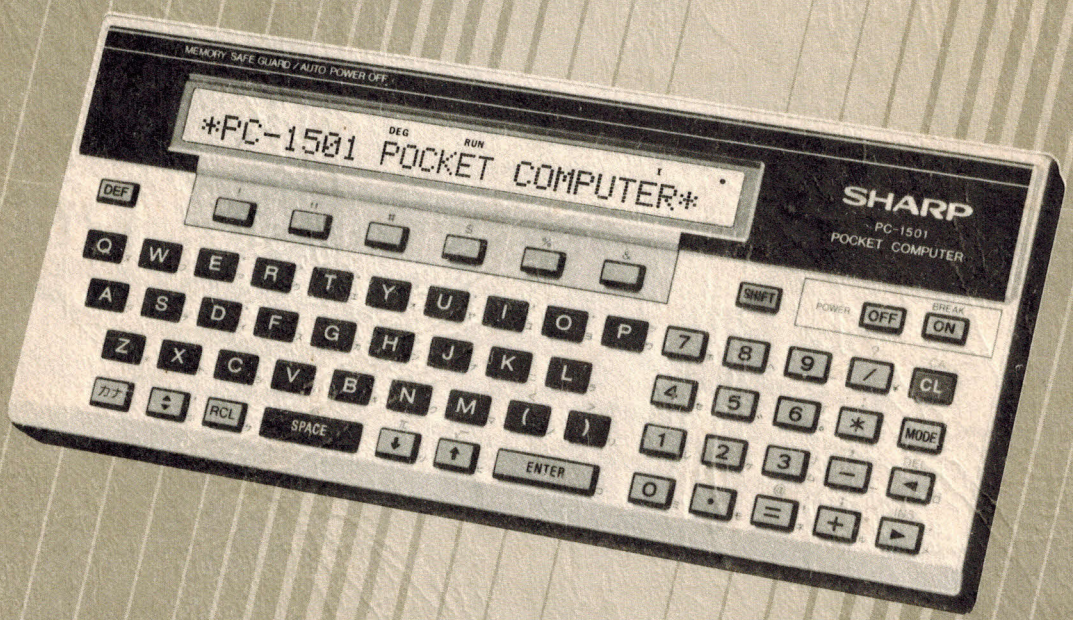
SHARP

ポケット コンピュータ

形名 PC-1501

取扱説明書

保証書付(巻末)
(WITH WARRANTY CARD)



信頼と保証のBMマーク
事務機械工業会認定品

〈はじめに〉

このたびはシャープポケットコンピュータ〈PC-1501〉をお買いあげいただき、まことにありがとうございました。

この計算機は、プログラム言語にパーソナルコンピュータなどで広く使用されていますBASIC(ベーシック)を用いたプログラム計算機です。

BASICは世界でもっとも多くの人が使用しているといわれるプログラム言語で、初めての人でも、少しの学習で容易にプログラムを組むことができます。

本機を正しくお使いいただくために、この取扱説明書をよくお読みください。

なお、この取扱説明書は、「シャープサービス・お客様ご相談窓口一覧表」とともに、必ず保存してください。万一ご使用中にわからないことが生じたとき、きっとお役に立ちます。

(ご注意)

- この取扱説明書の内容は改良のため予告なく変更する場合があります。
- 本機は非常に複雑な機能および組合せを有する製品であり、出荷に際して取扱説明書を含め十分なチェックをして万全を期しておりますが、万一ご使用中ご不審な点・お気づきのことがありましたらもよりのシャープサービス・お客様ご相談窓口までご連絡ください。なお運用した結果生じる影響については責任を負いかねますので、あらかじめご了承ください。
- 本書に記載のプログラムを使用したことによる金銭上の損害および逸失利益または第三者からのいかなる請求についても当社はその責任を負いませんのであらかじめ、ご了承ください。

●プリンタおよびテープレコーダーに関する命令について

本書84ページ以降に説明しておりますプリンタに関する命令(LPRINT、COLORなど)、およびテープレコーダーに関する命令(CHAIN、PRINT #など)は別売のカラーグラフィックプリンタ〈CE-150〉が持っている命令です。

したがって、本機(PC-1501)のみではこれらの命令をプログラムの中に用いることはできません。また、マニュアル操作で実行することはできません。

なお、CE-150と本機を接続しているときに、これらの命令を用いてプログラムを作成し、その後CE-150を取りはずした場合は、これらの命令は実行できません(エラーになります)のでご注意ください。

(再びCE-150を接続すれば使用できます。)

目　次

（ページ）

◆おねがい……4
◆本機の主な特長……5
◆電池の入れかた、交換のしかた……6

1．キー操作と表示……8
1－1）各部のなまえ……8
1－2）キー操作のしかた……8
1－3）モードについて……10

2．マニュアル計算……11
2－1）基本計算……12
2－2）関数計算……15
2－3）メモリー計算……18

3．プログラム(その1)……20
3－1）簡単なプログラム……21
（INPUT, LET, PRINT, END）
3－2）プログラムの入れかたと編集…23
（NEW, LIST）
3－3）プログラムの実行……25
（RUN）
3－4）いろいろな入力形……26
（INPUT）
3－5）いろいろな出力形……27
（PRINT）

4．プログラム(その2)……29
4－1）ジャンプと判断……29
（GOTO, IF…THEN）
(1) ジャンプ……29
(2) 判　断……30
4－2）くり返し計算……35
（FOR…TO…STEP, NEXT）
4－3）サブルーチン……37
（GOSUB, RETURN）
4－4）定義付けプログラム……39

5．変　数……42
5－1）変数の構成……42

（ページ）

5－2）配列変数（DIM）……42
5－3）メモリーの構成と変数……44
5－4）変数（メモリー）の内容を消すには（CLEAR）……46

6．表示に関するBASIC……47
6－1）いろいろな表示のさせかた……47
（PRINT, USING, CURSOR, WAIT, PAUSE, CLS）
6－2）文字や記号の作りかた……51
（GCURSOR, GPRINT, POINT）

7．そのほかのBASIC……55
7－1）READとDATA, RESTORE…55
7－2）ON GOTO, ON GOSUB……57
7－3）ON ERROR（エラー処理）…58
7－4）REM……58
7－5）AREAD……58
7－6）STOP……59
7－7）ARUN……59

8．特別な機能……60
8－1）乱数関数……60
8－2）TIME……61
8－3）INKEY$……61
8－4）音発生機能（BEEP）……62
8－5）CONT……63
8－6）STATUS, MEM……63
8－7）論理積、論理和、否定……64
（AND, OR, NOT）
8－8）条件式の結合……66
8－9）モードロック……67
（LOCK, UNLOCK）
8－10）OPN……68
9．デバッグ……68
（TR ON, TR OFF）
10．文字の編集……70
10－1）文字列の分断……70
（LEFT$, RIGHT$, MID$）
10－2）文字列の結合……72

（ページ）

10－3）文字数を数える方法（ＬＥＮ）… 72
10－4）数値と文字の変換…… 73
（ＶＡＬ，ＳＴＲ$）
10－5）アスキーコードの変換…… 73
（ＡＳＣ，ＣＨＲ$）
10－6）文字列の比較…… 74

11．リザーブ機能…… 76
11－1）リザーブのしかた…… 76
11－2）ソフトキーの
キーシンボル表示…… 79

◆カラーグラフィックプリンタについて…… 82
〈ＣＥ-１５０と計算機との接続〉 …… 82

●プリンタに関する命令…… 84
（１）モード指定命令…… 84
（ＴＥＸＴ，ＧＲＡＰＨ）
（２）プリント命令（ＬＰＲＩＮＴ）…… 84
（３）文字指定命令…… 85
（CSIZE，ROTATE，COLOR）
（４）色やインキのチェック（ＴＥＳＴ）…… 86
（５）印字位置の指定（ＴＡＢ）…… 86
（６）紙送り命令（ＬＦ）…… 87
（７）ペン移動命令（ＧＬＣＵＲＳＯＲ）…… 88
（８）原点指定命令（ＳＯＲＧＮ）…… 88
（９）描線命令（ＬＩＮＥ，ＲＬＩＮＥ）…… 89
（10）リストのしかた（ＬＬＩＳＴ）…… 92

〈テープレコーダーとの接続方法〉…… 94
●テープレコーダーに関する命令…… 96
（１）プログラムの記録命令…… 96
（ＣＳＡＶＥ）
（２）プログラムの転送命令…… 97
（CLOAD，MERGE，CHAIN）
（３）照合命令（ＣＬＯＡＤ？）……101
（４）データの記録命令（PRINT #）……
（５）データの転送命令（INPUT #）……102
（６）リモートコントロール機能について…103
（RMT OFF，RMT ON）

◆計算機とテープレコーダーの操作方法……104

（ページ）

（１）準　備……104
（２）テープへの記録方法……105
（３）計算機とテープの内容の照合……107
（４）テープからの転送……108
（５）ＣＥ-１５２以外のテープレコーダーを
使用する場合について…… 110

◆キーの機能…… 111
◆オプション機器のご案内…… 115
◆仕　様…… 117

プログラム事例集…… 119
エラーコードとエラー内容…… 167
キャラクタ・コード表…… 170
関数・命令（省略形）一覧…… 171
索　引 …… 181

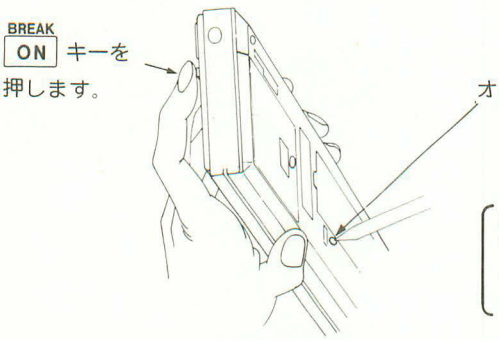
付属のネームラベルにお名前をご記入の上、計算機の裏面にはってご使用ください。

おねがい

- 本機は精密部品で構成されていますので、直射日光の当たる場所（特に車の中では非常に温度が高くなることがあります）や、暖ぼう器具の近くなど、高温になることがある場所に置かないでください。
  また、急激な温度の変化、低温、湿気、ホコリ、曲げ、ヒネリ、強いショックなどはさけておつかいください。故障などの原因になるおそれがあります。
- 液晶表示部はガラスでできていますので、強く押さえないでください。割れることがあります。
- お手入れの際は乾いた柔らかい布を使用し、ベンジン、シンナーなどの揮発性の液体や、ぬれぞうきんなどを使用しないでください。
  揮発性の液体やぬれぞうきんなどは計算機をいためることがあります。
- 電池を火中に投入しないでください。火中に投入すると破裂するおそれがあります。
- この計算機を分解しないでください。万一故障のときは、お買上げの販売店あるいは「シャープサービス・お客様ご相談窓口一覧表」に記載のもよりのサービス会社へ本書をそえてご持参ください。

異常が発生した場合の処理について

本機をご使用中に強度の外来ノイズや強いショックをうけた場合などに、ごくまれに [BREAK ON] キーも含めた、全てのキーが働かなくなるなどの異常が発生することがあります。
本機にこのような異常が発生した場合は、[BREAK ON] **キーを押さえたままで計算機裏面のオールリセットスイッチを約15秒間押してください。**



（オールリセットスイッチを操作するときはボールペンなどで押してください。しん先の出たシャープペンシルなど、先の折れやすいもの、また、とくに針などの先のとがったものは使用しないでください。）

そののち、表示部に [NEW 0 ? : CHECK] と表示されることを確認して

[CL] NEW 0 [ENTER]　（0は数字のゼロを示します。11ページ参照）

と押して、計算機内をクリア（消去）し、続いて

[SHIFT] [MODE] NEW [ENTER]

と押してリザーブメモリーの内容も消去してください。（リザーブについては76ページを参照してください。）

なお、オールリセットスイッチを押したのち、もしもNEW 0 ? : CHECK と表示されなかった場合は、もう一度上記方法でオールリセットスイッチを押してください。

（注）オールリセットスイッチを押しますと、プログラム、データ、リザーブ内容およびTIMEの内容（61ページ参照）が壊れますので、異常状態が発生したとき以外は、このスイッチを押さないでください。

本機の主な特長

★自由にシンボルを描くミニグラフィック表示

縦7ドット、横156ドットのマトリックス液晶表示部には、ミニグラフィック機能により自由にシンボルを表示させることができます。

ゲーム用シンボルやカタカナ、ひらがな、簡単な漢字など、目的に応じてプログラミングすれば楽しい、そしてスマートな表示をおこなわせることができます。しかも、この表示部は文字・数値表示用としてはワイドな26桁を表示させることができます。

★大きなメモリー容量、しかもメモリー保護機能つき

内部にはROM 16Kバイト、RAM 8.5Kバイトを標準実装。さらに別売のメモリーモジュールを取りつけることにより、RAM 4Kバイト、8Kバイト、16Kバイトなどの拡張ができます。この大きなメモリー容量により、まさにパソコンに迫るアプリケーションがポケットコンピュータで可能になりました。しかもプログラムやデータは電源を切っても保護されていますので使うたびに記憶させる必要がなく、いつでも、どこでも、すぐに使用できます。

ROM：Read Only Memory 読み出し専用メモリー

RAM：Random Access Memory 読み書きができるメモリー

バイト：メモリーの大きさやプログラムの大きさを表わす場合などに用いる単位

★高速ＣＰＵ採用で処理時間を大幅に短縮

当社独自開発による高速CMOS CPUの採用により、処理速度が一挙に約10倍（当社比）と高速化に成功!

これまで多くの時間を費やしていた複雑な技術計算、事務計算なども処理時間が大幅に短縮され、効率的に使用することができます。

ＣＰＵ：Central Processing Unit 中央処理装置

★自由に設計できるソフトウェアキー

6個のソフトウェアキー（ソフトキー）は表示と対で使用し、キー操作をリザーブするリザーブキーとして、あるいはプログラムの頭出しができる定義付けキーとして、さらには表示に対してのアクションをおこなうゲーム用キーとして、プログラムにより自由に設計し、使用することができます。

しかも3段切換式になっており、合計18個のソフトウェアキーとして使用することができます。（76ページ参照）

★ＰＣ-1500のプログラム、周辺機器が使えます

本機は、発売以来好評をいただいておりますPC-1500のRAM容量を大きくし、より大きなプログラムの作成・実行を可能にしたポケットコンピュータです。

したがって、これまでPC-1500で開発、蓄えられたプログラムをそのまま使用することができます。

また、PC-1500の周辺機器はすべて使用することができます。

電池の入れかた、交換のしかた

電池の交換時期

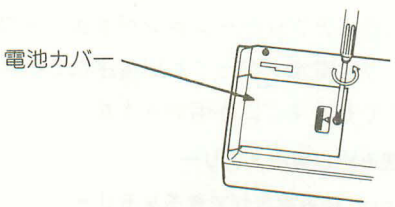
正面から見て、表示部右端にあるバッテリーインジケーター「●」がうっすらとしか見えなくなった場合は電池の消耗を示していますので、速やかに新しい電池と交換してください。

（別売の周辺機器ＣＥ-150とＣＥ-152をお持ちの場合は、事前にプログラムやデータをテープに記録してください。）

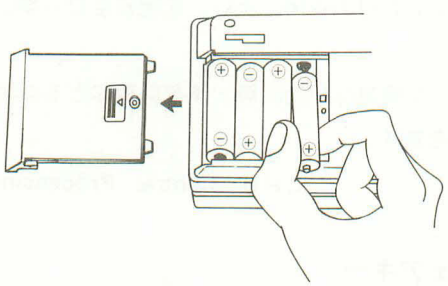
電池を入れる場合、および交換する場合は次の手順にしたがっておこなってください。

① [OFF] キーを押して電源を切ってください。

② 硬貨や小型のネジ回しで電池カバーのビス（ネジ）をはずし、電池カバーをはずしてください。



③ 使用ずみの電池を取りはずし、単３タイプの新しい乾電池を４本、電池の⊕・⊖をまちがえないように入れてください。



④ 電池カバーをはめてビスで留めてください。（ビスはゆるまないように締めてください。）

⑤ （ [ON] ） [CL] ＮＥＷ Ø [ENTER]

と押して計算機内をクリア（消去）したのち、次のようにプロンプト記号が表示されることを確認してください。

```
>
```

└プロンプト記号が表示されることを確認してください。

もし、上記以外の表示がなされているときは、電池を取りはずし、改めて入れたのち、４ページの方法でオールリセットスイッチを押してください。

（注） 表示部に　NEW0?　：CHECK　と表示されたときは、必らず

[CL]　NEW　0　[ENTER]

と操作してください。

もし、上記操作をおこなわなかった場合、メモリー内や計算機の状態が一定せず、プログラムが書き込めないことや、計算機が迷走して止まらなくなることがありますので、必ず操作してください。

もし、迷走して止まらなくなった場合は4ページの方法でオールリセットスイッチを押してください。

なお、上記操作をおこなっても、リザーブメモリーの内容は消去されず、不確定な内容が残る場合がありますので、76ページの「準備」で説明の方法で内容を消去してください。

◆この計算機には単3タイプの乾電池を4本ご使用ください。

〈乾電池取扱上のご注意〉

電池は誤った使い方をしますと、液もれや、破れつすることがありますので、次の点について特にご注意ください。

① 電池のプラス⊕とマイナス⊖の向きを本機の表示通り正しく入れてください。

② 新しい電池と、一度使用した電池を混ぜて使用しないでください。

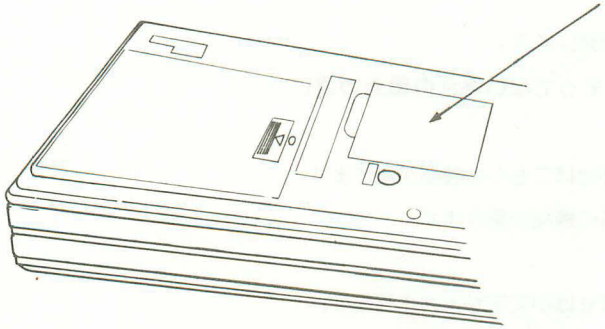
③ 種類のちがう電池を混ぜて使用しないでください。同じ形状でも電圧の異なるものがあります。

④ 電池は使えなくなったり、長時間使わないときは本機から取り外しておいてください。

⑤ 電池には充電式と充電式でないものがあります。電池の注意表示をよく見てご使用ください。

〈ご注意〉

- 消耗した電池をそのままにしておきますと、電池の液がもれて計算機をいためることがあります。
- 電池を火中に投入しないでください。火中に投入すると破裂するおそれがあります。
- 本機には充電式電池は使用できません。(また、充電もできません。)

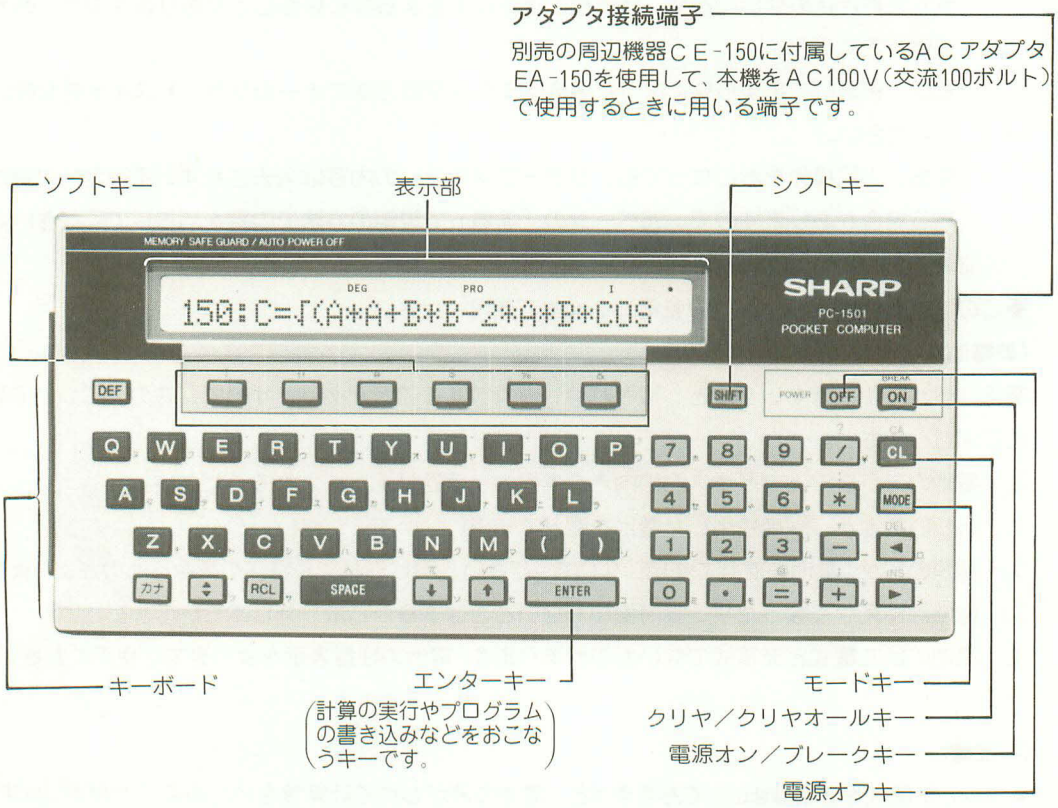


モジュールボックス
別売のメモリーモジュールを取りつけるところです。

ボックス内の端子(コネクタ)に、手などで触れないでください。
静電気などにより、本機内に悪影響をおよぼす恐れがあります。

1．キー操作と表示

1－1）各部のなまえ



1－2）キー操作のしかた

本機はキーがたくさんあります。(65キー)

各キーはそのまま押す場合と、SHIFT キー(シフトキー)に続いて押す場合とがあり、それぞれの場合で働く機能が違います。

ここではキーの操作について簡単に説明します。

(1) **シフトキーに続いて押した場合とそうでない場合の働きの違い**

(例) SHIFT キーに続いて押せばこちらの機能が働きます。

DEL ◄ ←そのまま押せばこちらの機能が働きます。

SHIFT キーに続いて押せば小文字が入力されます。

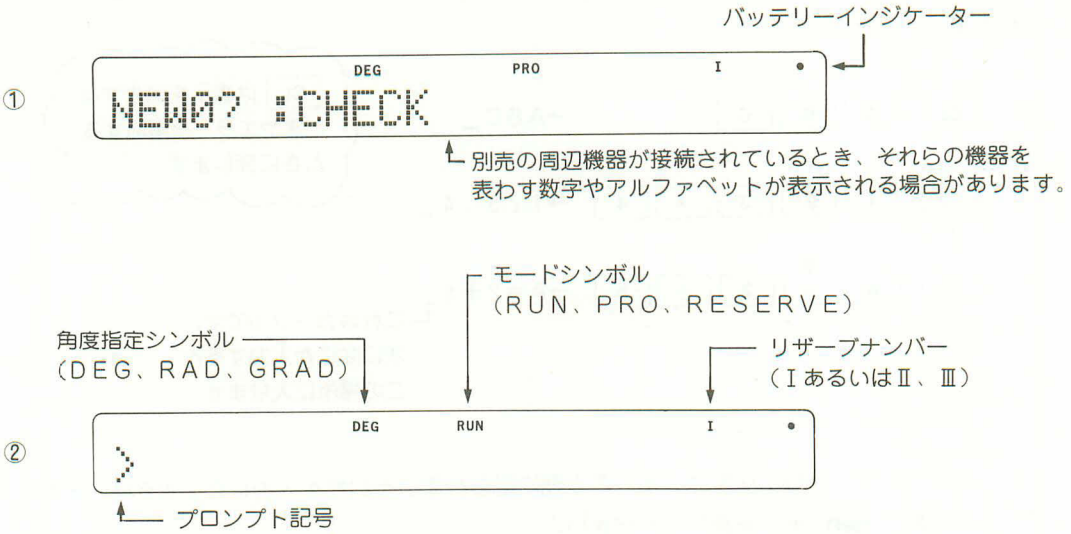
A ←そのまま押せば大文字が入力されます。

SHIFT キーに続いて押せばキーの上側に記されている記号が入力されます。

; + ←そのまま押せばキーに直接記されている記号が入力されます。

(2) 電源のオン／オフ

まず始めにキーボード右上にある [ON] キーを押してください。電源が入り、次のような表示になります。



①の表示は電池を交換したときなどに表示されます。もし、①の表示になったときはまず最初に次のように操作して計算機内をクリアしてください。

[CL] NEW 0 [ENTER]

NEW 0 は“すべての状態を解除(初期状態に)しなさい”という命令です。

この操作をおこなえば、表示は②のように、プロンプト記号が表示されます。

[OFF] キーで電源を切ったのち、[ON] キーで電源を入れれば、②の表示になります。

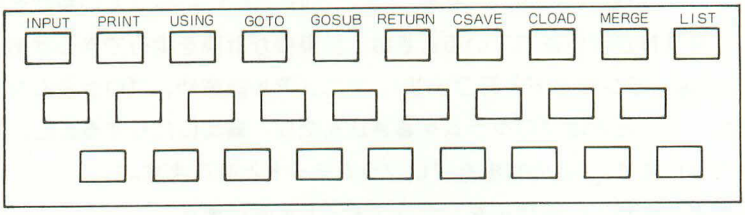
(ただし、別売の周辺機器が接続されているとき、その周辺機器になんらかの異常がある場合は、たとえば ：CHECK 6 のようにチェックメッセージが表示されます。)

●オートパワーオフ（AUTO POWER OFF）

本機はキー入力が可能なときに、約7分間新たなキー操作をおこないませんと、電池の消耗を少なくするため、自動的に電源が切れます。

この場合、[ON] キーを押せば電源が入りますが、計算機の状態、表示は電源が切れる前の状態にもどります。

テンプレート



テンプレートは計算機のキーボードのアルファベットキー部分に載せて使用します。アルファベットキーは定義付けキーとしてプログラムなどを定義付けることができますが、各キーに定義付けしたプログラムを表わす記号や文字をテンプレートに書いて利用してください。(定義付けプログラムについては 39 ページを参照)

(3) アルファベットや記号、数字の入力

電源を入れたのち、[A] ～ [Z]、[0] ～ [9]、[•]、[?/] [:*] [,-] [;+] [@=]、[<(] [>)] などのキーを押せば、それぞれのキーに直接記されている文字や記号、数字が入力されます。

(例) [CL] [A] [B] [C] →ABC_

[CL] [1] [2] [3] [•] [4] →123.4_

[CL] [A] [@=] [2] [;+] [5] →A=2+5_

[CL]は表示を消去するときやエラーを解除するときに押します。

_ これはカーソルです。次になにか入力するとこの場所に入ります。

もし、アルファベットの小文字や、キーの上側に記されている記号を入力したいときは、それぞれのキーの前に [SHIFT] キーを押してください。

(例) [CL] [SHIFT] [A] [SHIFT] [B] [C] →abc_

[CL] [SHIFT] [?/] [SHIFT] [;+] [SHIFT] [#] →?;#_

(注)本機のキーの大部分は横にカタカナや記号が記されていますが、これらの文字や記号および [カナ] キーは別売の「カタカナ」があるときにのみ入力することができます。

1－3）モードについて

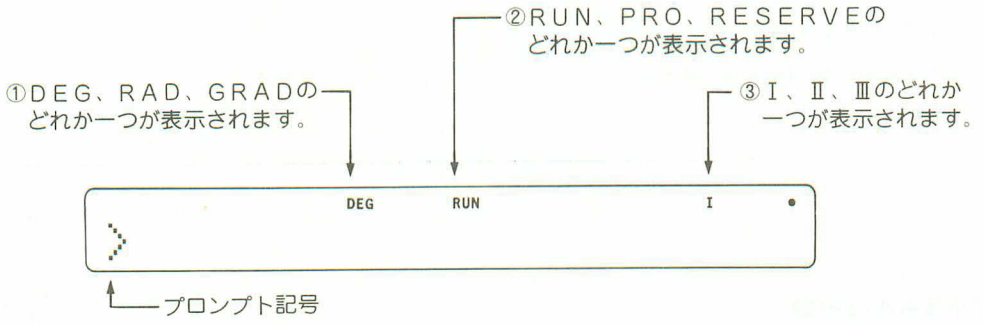
電源を入れてキーボードの右端の列にある [MODE] キーを何回か押してみてください。
表示部の上のほうでRUNとPROというシンボルが交互に表示されることがわかります。
次に [SHIFT] [MODE] と押してください。今度はRUNやPROが消えてRESERVEが表示されます。(もし、表示が替わらないときは、UNLOCK [ENTER] と操作してから押してください。)
これらのシンボルは計算機がどのような状態(モードと呼びます)になっているかを示すシンボルです。すなわち、RUNが表示されているときは、計算機が計算を実行できる状態になっていることを示し(この状態をランモードと呼びます)、マニュアル計算や、プログラムを実行します。
PROが表示されているときはプログラムを書き込んだり、編集したりすることのできる状態になっていることを示します。(この状態をプログラムモードと呼びます。)
RESERVEが表示されている状態をリザーブモードと呼びます。
リザーブモードは、ソフトウェアキーにキー操作などをリザーブするモードです。
(リザーブについては76ページで説明)

2．マニュアル計算

ここでは、計算をマニュアルでおこない、本機の操作になれていただくことと、いろいろな基本計算をマスターしていただくことを目的とします。

準備１．電源オン

まず [ON] キーを押してください。電源が入り、表示部の左端に 〉（プロンプト記号）が表示されます。



準備２．モード指定

[MODE] キーを操作しますと、RUNとPROが交互に表示されますので、マニュアル計算のときはRUNを表示させてください。（ランモードを指定してください。）

もし、[MODE] キーを押してもRUNとPROの表示が替わらないときは、UNLOCK [ENTER] と操作してから、[MODE] キーを押してください。

＝注＝

本書ではキー操作やプログラムの説明において、数字ゼロ（０）を　Ø　と表わしている場合があります。Øは表示あるいは印字において、アルファベットのオー（Ｏ）と区別するための数字で、本書においても、オーとゼロの区別がつきにくい場合はゼロを　Ø　と表わしています。

2－1）基本計算

〔**例題**〕次の計算をおこなってください。

(1) $2+3\times4=$

(2) $5\times(-6)=$

(3) $5\times(-6)+7=$

(4) $36\div(1+2)=$

(5) $\dfrac{4+5}{2+3}=$

(6) $1+2(-3+4(5+6(7-8)))=$

(7) $5\times10^{3}\div(4\times10^{-3})=$

(8) $-1.2\times(-4)+(-5)\times6=$

(9) $2\pi\times5=$

〔**解答**〕

(1) ２＋３＊４ ENTER　　１４

(2) ５＊－６ ENTER　　－３０

(3) ５＊－６＋７ ENTER　　－２３

(4) ３６／（１＋２） ENTER　　１２

(5) （４＋５）／（２＋３） ENTER　　１．８

(6) １＋２＊（－３＋４＊（５＋６＊（７－８））） ENTER　　－１３

(7) ５Ｅ３／４Ｅ－３ ENTER　　１２５０００

(8) CL －１．２＊－４＋－５＊６ ENTER　　－２５．２

(9) ２＊π＊５ ENTER　　３１．４１５９２６５４

　　（π は SHIFT ↓ と押します。）

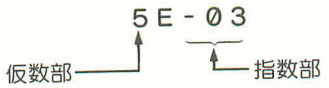
〔**説明**〕

◎ **基本キー**

通常の数学では四則計算の記号として「＋，－，×，÷」を使いますが、本機では「×や÷」は使わず、「＊と／」を使います。これらを次にまとめます。

加える（加算）	＋
引　く（減算）	－
かける（乗算）	＊（アスタリスク）
割　る（除算）	／（スラッシュ）
符　号	－または＋
指　数※	Ｅ（例題の(7)を参照）
計算の実行	ENTER

※20000を 2×10^4、0.005を 5×10^{-3} のように表わす方法を浮動小数点表示(指数表示)と呼びます。
これらを本機で表わす場合は指数部を示すEを入れて2E　04、5E−03　のような形で表示します。また、キー入力する場合は　2E4、5E−3　のようにキーを押します。



なお、指数部に3桁以上の数値を入れた場合は、後2桁の数字が有効になります。

（例）　5E123 [ENTER] →5E　23

◎　**計算の優先順位**

本機ではカッコ、関数も含めて数式通りのキー操作で計算をおこなうことができます。計算の優先順位の判断や途中結果の処理はすべて計算機が自動的に処理してくれます。計算の優先順位は次の通りです。

1. 関　数（sin，cosなど）
2. べき乗（^）　（(例)　5^3=5^3、3^−2=3^{-2}）
3. 符　号
4. 乗除算
5. 加減算
6. 論理演算（>，>=など）
7. AND，OR

注）●カッコが使用されている場合はカッコ内の計算が最優先されます。
●複合関数（sin cos^{-1} 0.6など）は右から左の順で計算されます。
●べき乗の連算（3^{4^2} 即ち 3^4^2 など）は右から左の順で計算されます。
●2．と3．では後に出てきたほうが優先順位が高くなります。

（例）　−2^4　→　−(2^4)
　　3^−2　→　3^{-2}

◎　**数式の長さ**

一つの数式の長さ（ [ENTER] を押すまで）は80キー操作まで可能です。

◎　**キー操作の訂正**

もしも誤ってキー操作をしてもあわてずに次の方法で訂正しましょう。

(1)　誤りが多いとき

[CL] キーを押せば表示が消えて、左端に〉記号が表示されますので、最初から入れ直します。

(2)　誤って他のキーを押したとき

誤って押したキーのシンボルが、点滅表示されるまで [◄] キーを押して、そこで正しいキーを押します。（ [►] や [◄] は押し続けるとカーソル※が続けて移動します。）

（例）　2＋3＊4を2＋3／4とした場合は

2＋3／4
　└この部分を点滅させて＊を押し、 [ENTER] キーを押します。

※カーソル

キー操作をおこなうとき_を表示したり、キー操作の訂正のときに点滅表示をしたりしますが、これらは次にキー操作をしたキーがこの部分に入ることを示しており、これをカーソルと呼びます。

(3) 誤って余分なキーを押したとき

余分なキーのシンボルが点滅表示されるまで ◄ キーを押し、SHIFT DEL と押します。(DEL：DELETE(デリート)…削除を意味します。)

(例) 2＋3＊4 を 2＋3＊／4 とした場合は

◄ を操作　　2＋3＊／4
　　└この部分を点滅させてください。

SHIFT DEL と押す　　2＋3＊4
　　└ここが点滅します。

ENTER を押す　　14　(答)

(4) キーの操作がぬけたときは

ぬけたキー操作部分の次のキーシンボルが点滅するまで ◄ キーを押し、そこで SHIFT INS と押すと一文字分の空欄ができますので、次にぬけていたキーを押せば、その位置に入力されます。(INS：INSERT(インサート)…挿入を意味します。)

(例) 2＋3＊4 を 2＋34 とした場合は

◄ を操作　　2＋34
　　└この部分を点滅させてください。

SHIFT INS と押す　　2＋3□4
　　└ここが点滅します。

＊ を押す　　2＋3＊4
　　└ここが点滅します。

ENTER を押す　　14　(答)

◎ プレイバック機能

ENTER キーで計算を実行したのち、◄ あるいは ► キーを押すと実行した式が呼びもどされます。これは実行した式の確認や実行した式の一部を変更して再度実行したいときに便利です。特に ENTER で計算を実行した結果、エラーになったときにこのプレイバックを使いますと、エラーを起こした場所をカーソルで示していますので、エラーの原因がわかりやすく大変便利です。(>記号(プロンプト記号)が表示されたときには、すでに内容が消されていますのでプレイバックできません。)

(例) 36／1＋2) ENTER と押してエラーになったとき

◄ を押すと　　36／1＋2)
　　└点滅表示

◎ **連続計算のしかた**

[ENTER] キーを押しますとその計算結果が表示されますが、その結果を続けて計算することもできます。

（例） 1＋2 [ENTER] 3
＊4 [ENTER] 12（3×4を実行）

◎ **エラー表示**

マニュアル計算でエラーになりますと

ERROR 1 や ERROR 37

のようにエラー番号が表示されます。（エラー番号の意味は167ページの「エラーコードとエラー内容」を参照ください。）

2－2）関数計算

〔例 題〕次の計算をおこなってください。

(1) $\sin 30^\circ + \cos 60^\circ \times \tan 45^\circ =$

(2) $\tan \frac{\pi}{4} =$

(3) $\sin(15^\circ + 30^\circ \div 2) =$

(4) $\sin(-30^\circ) =$

(5) $\sqrt{3^2+4^2} =$

(6) $\sin^2 30^\circ =$

(7) $\log 5 + \ln 5 =$

(8) $e^{2+3} =$

(9) 60進数の30度30分は10進数の何度？

(10) 10進数の30.5度は60進数の何度何分何秒？

〔解 答〕(1)(3)(4)(6)では角度指定をＤＥＧに、(2)では角度指定をＲＡＤにして計算。

（次ページ参照）

(1) ＳＩＮ３０＋ＣＯＳ６０＊ＴＡＮ４５ [ENTER] 1

(2) ＴＡＮ（π／４） [ENTER] 1

(3) ＳＩＮ（１５＋３０／２） [ENTER] ０．５

(4) ＳＩＮ－３０ [ENTER] －０．５

(5) √（３へ２＋４へ２） [ENTER] ５

(6) ＳＩＮ３０へ２ [ENTER] ０．２５

(7) ＬＯＧ５＋ＬＮ５ [ENTER] ２．３０８４０７９１７

(8) ＥＸＰ（２＋３） [ENTER] １４８．４１３１５９１

(9) ＤＥＧ３０．３０ [ENTER] ３０．５（30.5度）

(10) ＤＭＳ３０．５ [ENTER] ３０．３（30度30分）

〔説　明〕

◎　角度指定

三角関数、逆三角関数では、計算する角度の単位を正しく指定しておく必要があります。

角度単位	表示シンボル	指定のしかた	備　　考
度	**DEG**	DEGREE [ENTER]	直角を90で表わす単位〔°〕
ラディアン	**RAD**	RADIAN [ENTER]	直角を$\frac{\pi}{2}$で表わす単位〔rad〕
グラード	**GRAD**	GRAD [ENTER]	直角を100で表わす単位〔^{g}〕

（$90^\circ=\frac{\pi}{2}$〔rad〕$=100^{g}$）

●指定された角度単位はシンボルとして表示部に表示されます。（DEG、RAD、GRAD）

◎　関数機能

項　　目	一般の表示	本機のキー操作	備　　考
三角関数	sin cos tan	SIN COS TAN	
逆三角関数	$\sin^{-1}$ $\cos^{-1}$ $\tan^{-1}$	ASN ACS ATN	
常用対数	log	LOG	$\log_{10} x$（10を底とする対数）
自然対数	ln	LN	$\log_e x$（eを底とする対数）
指数関数	e^x	EXP	e＝2.718281828
べき乗		^	A^BにおいてA^B
開平	√	√あるいはSQR	
10進数(度)→60進数(度分秒)変換		DMS	角度変換
60進数(度分秒)→10進数(度)変換		DEG	
整数化（インテジャー）		INT	INT(x)において、x以下でかつもっとも大きい整数を求めます。
絶対値（アブソリュート）	|X|	ABS	ABS(x)において、xの絶対値を求めます。
符号関数（シグナム）		SGN	SGN(x)において、x＞0のとき1、x＜0のとき−1、x＝0のとき0が得られます。
円周率	π	πあるいはPI	π≒3.141592654

〈関数の計算範囲〉

関　　数	計　算　範　囲
$y \wedge x$　（yのx乗）	$-1\times10^{100}<x\log y<100$ ｛$y=0$、$x\leqq0$のときはERROR 39 $y=0$、$x>0$のときは0 $y<0$、$x\neq$整数のときはERROR39｝下記の例を参照
SIN x COS x TAN x	DEG：$\lvert x\rvert<1\times10^{10}$ RAD：$\lvert x\rvert<\frac{\pi}{180}\times10^{10}$ GRAD：$\lvert x\rvert<\frac{10}{9}\times10^{10}$ ｝ただし、TANxにおいて次の場合は除く DEG：$\lvert x\rvert=90(2n-1)$ RAD：$\lvert x\rvert=\frac{\pi}{2}(2n-1)$ GRAD：$\lvert x\rvert=100(2n-1)$（nは整数）
ASN x ACS x	$-1\leqq x\leqq1$
ATN x	$\lvert x\rvert<1\times10^{100}$
LNx LOGx	$1\times10^{-99}\leqq x<1\times10^{100}$
EXPx	$-1\times10^{100}<x\leqq230.2585092$
$\sqrt{x}$	$0\leqq x<1\times10^{100}$

上記以外の関数は、演算数xが次の範囲で計算可能です。

$1\times10^{-99}\leqq\lvert x\rvert<1\times10^{100}$ および0

（例）
0 ^ 0 [ENTER] → ERROR 39
0 ^ 5 [ENTER] → 0
(−4) ^ 0.5 [ENTER] → ERROR 39
−4 ^ 0.5 [ENTER] → −2

●関数の精度は原則として10桁目に±1（指数表示の場合は仮数部の最下位桁に±1）の誤差となります。

2－3）メモリー計算

〔例　題〕

(1) ㋑Aメモリーに3，Bメモリーに4を入れてください。
㋺$\sqrt{A^2+B^2}$を計算してその結果をCメモリーに入れてください。
㋩Aメモリー、Bメモリー、Cメモリーの内容を順次確認してください。

(2) ㋑上のAメモリーの内容に1を、Bメモリーの内容に2を加えて、それぞれのメモリーに入れてください。
㋺$\sqrt{A^2+B^2}$を計算して結果をCメモリーに入れてください。
㋩A、B、Cの各メモリーの内容を順次確認してください。

(3) ㋑A$メモリーに“PC-1501”、B$メモリーに“ＢＡＳＩＣ”という文字を入れてください。
㋺A$メモリー、B$メモリーを呼び出して確認してください。

〔解　答〕

(1) ㋑A＝3 ENTER
B＝4 ENTER
} またはA＝3，B＝4 ENTER
㋺C＝√(A＊A＋B＊B) ENTER
㋩A ENTER　3
B ENTER　4
C ENTER　5

(2) ㋑A＝A＋1 ENTER
B＝B＋2 ENTER
㋺C＝√(A＊A＋B＊B) ENTER
㋩A ENTER　4
B ENTER　6
C ENTER　7．211102551

(3) ㋑A$＝"ＰＣ－１５０１" ENTER
B$＝"ＢＡＳＩＣ" ENTER
㋺A$ ENTER　ＰＣ－１５０１
B$ ENTER　ＢＡＳＩＣ

〔説　明〕

メモリーについてのくわしい説明は42ページでの説明にゆずり、ここでは数値や文字を入れておくデータメモリーについて簡単に説明します。

データメモリーは次のように分類されます。（パーソナルコンピュータなど、上級機ではデータメモリーのことを変数と呼ぶのが普通です。）

データメモリー（変数）{ 数値メモリー（数値変数）
文字メモリー（文字変数）

◎ 数値変数への数値の格納（代入）

A＝3 ENTER

A＝A＋1 ENTER

のように、数値変数＝式 ENTER と操作します。

変数の名前はA～Zだけでなく、AA、BZ、C5などのように2個の文字・数字を名前に使用することができます。この場合は、最初の文字が必ず英文字（大文字）でなければなりません。（ただし、PIやIFなどは計算機が関数などの命令と判断しますので使用できません。）

（例） AA＝5 ENTER

ZC＝7＋2 ENTER

A1＝AA＊ZC ENTER

変数の名前に3個以上の文字・数字を使用してもかまいません。ただし、最初の2字以外は無視されます。

たとえば変数名にABCやKOTAEなどを使用することができますが、KOTAE1とKOTAE2では計算機には区別できません。

（最初の2字以外は無視されますので、計算機はKOという同じ変数と判断してしまいます）

●数値変数には仮数部10桁、指数部－99～99までの数値を格納することができます。

◎ 文字変数への文字の格納

変数名の後に$（ドルマーク）をつけることにより文字変数を表わし、

文字変数＝“文字” ENTER

または

文字変数＝文字変数 ENTER

と操作します。

（例） A$＝"SHARP" ENTER

BZ$＝"COMPUTER" ENTER

C3$＝A$ ENTER

●文字変数には最大16文字まで格納することができます。

3．プログラム(その１)

前項でいろいろなマニュアル計算に慣れていただきましたが、いろいろな計算の中でまったく同じような計算を何度もおこなうことがよくあります。

たとえば、円の面積を求める計算を例に取ってみましょう。円の面積Ｓを求める式は

$S=\pi r^2$

でしたね。そしてｒが半径です。

半径が３㎝の円の面積は？

π＊３＊３ [ENTER]

半径が４㎝では？

π＊４＊４ [ENTER]

などと操作します。このように何度も使う計算の手順を計算機に覚えさせておけば、上例のｒの値のように、その都度変わる数を入力するだけで答が一度に求められるのです。

さらにいろいろな判断が必要になる複雑な計算も、その手順を計算機に覚えさせておけば、人間が順次判断するのと同様に計算機が判断して一度に答を求めてくれます。

計算の手順を計算機が理解できるように書き表わしたり、計算機に覚えさせることをプログラミングとかプログラムを組むなどと言います。

ここではプログラムの基本について説明します。

●プログラムを計算機に覚えさせるときはＰＲＯモードにしてください。

プログラミングの前に

電池交換をおこなった後などでは、計算機の状態などが一定せず、プログラムが書き込めないことがあります。したがって、はじめてプログラミングをおこなうときは最初に

[CL] ＮＥＷ ０ [ENTER]

と押してください。

また、古いプログラムを消すときは

（[CL]） ＮＥＷ [ENTER]

と押してください。

3－1）簡単なプログラム（INPUT、LET、PRINT、END）

〔例　題〕

半径を入力して円の面積を求めるプログラムを作ってください。

〔解　答〕（プログラム１）

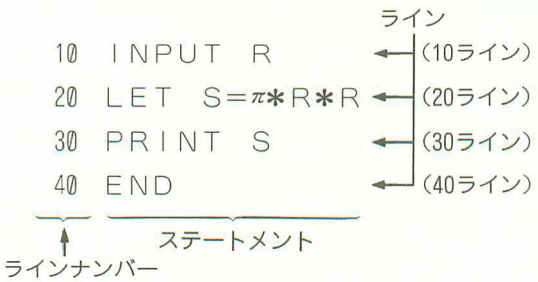
```
10  INPUT  R
20  LET  S=π*R*R        ←通常、ＬＥＴは省略できます。
30  PRINT  S
40  END
```

〔説　明　1〕

ＢＡＳＩＣをまったく知らない人でもこのプログラムが計算機に何を伝えているのか想像がつくと思います。これは次のような指示をするプログラムなのです。

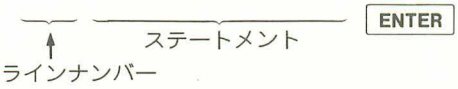
```
10  INPUT  R          ←データを読み込み、Rに入れなさい。
20  LET  S=π*R*R      ←πにRの値を２回掛けなさい。そして、その結果をSに入れなさい。
30  PRINT  S          ←Sの値を表示しなさい。
40  END               ←これでプログラムは終りです。
```

さて、ここでプログラムの構成と各命令について少しくわしく説明します。プログラムの一番基本となるところですから十分理解するようにしてください。



◎　ライン

ＢＡＳＩＣでは、プログラムはいくつかのラインから構成されています。そして１つのラインは、1つのラインナンバーと、いくつかの意味を持ったステートメント（命令など）からなります。（各ラインの終りは ENTER キーで指定します。）



（ ENTER はラインの終りを意味し、計算機にプログラムを入れるときには必ず必要です。）

◎ **ラインナンバー**

ラインナンバーは１～65279の整数をつけます。

プログラムの実行は若いラインナンバーから順次実行されますので、計算の順に若いラインナンバーをつけます。ただし、あとからプログラムの途中にラインを追加したい場合も出てきますので、そういったことを配慮して10、20、30……のように10番おきにラインナンバーをつけるのが最も一般的です。

◎ **ステートメント**

１つの意味を持った処理式や命令をステートメントといいます。

１つのラインは１つ以上のステートメントからなり、２つ以上のステートメントになる場合はステートメントとステートメントの間に：(コロン)を入れて区別します。

ラインナンバー　ステートメント：ステートメント：ステートメント [ENTER]

◎ **ＩＮＰＵＴ命令（インプット）**　（くわしくは、26ページで説明します。）

プログラム実行中にマニュアル操作により数値や文字の入力をおこなうための命令で、与えられた数値や文字を入れる数値変数や文字変数をＩＮＰＵＴに続いて書きます。

```
INPUT　A　（変数Aに数値を入れなさい）
INPUT　C$（変数C$に文字を入れなさい）
```

なお、複数の数値や文字を連続して入れることもでき、次のように使います。

```
INPUT　A，B
INPUT　C$，D$
INPUT　A，B，C$
```

◎ **ＬＥＴ命令（レット）**

代入文を意味し、ＬＥＴに続いて代入式を書きます。

```
LET　S＝π＊Ｒ＊Ｒ
```

なお、ＬＥＴは通常省略して使用できます。代入式は一般の数学とは少し異なり、右辺の数値あるいは数式を計算して左辺の変数に入れる(代入する)ことを意味します。したがって左辺は変数に限ります。

◎ **ＰＲＩＮＴ命令（プリント）**　（くわしくは、27ページで説明します。）

ＰＲＩＮＴに続いて変数を書いて、その変数の内容を表示させます。

```
PRINT　A　（Aの値が表示部の右端につめて表示されます。）
PRINT　A$（A$の内容が表示部の左端から表示されます。）
```

なお、変数の替わりに式や文字を書くこともできます。

```
PRINT　2＋3
PRINT　"KOTAE"
```

◎ **ＥＮＤ命令（エンド）**

ＢＡＳＩＣではプログラムの終りに“これでプログラムは終了です！”という意味でＥＮＤ文を書きます。

3－2）プログラムの入れかたと編集（NEW、LIST）

◎ **プログラムの入れかた（書き込み）**

先に作りましたプログラム(プログラム１)を計算機に書き込み、覚えさせましょう。

○まず、[MODE] キーを押して、表示部にPROシンボルを表示させてください。(PROモードにします。)

○次に、NEW [ENTER] と押してください。これで、先に記憶されている（かもしれない）プログラムやデータをすべて消します。

○続いてプログラムを１ラインずつ順番に入れてください。

このとき注意していただきたいことは、１ラインを入れ終るごとに [ENTER] キーを押すことです。

たとえば 10 INPUT R [ENTER] と押します。

[ENTER] は“このラインは終りましたョ”という意味で押します。

このように、１ライン入れ終るごとに [ENTER] を押すことを忘れないで、すべてのプログラムを入れてください。

[MODE] ← PROシンボルを表示させます。

NEW [ENTER]

10 INPUT R [ENTER]

20 LET S＝π＊R＊R [ENTER]

30 PRINT S [ENTER]

40 END [ENTER]

プログラムを入れていただければ、おわかりと思いますが、各ラインを入れ、[ENTER] と押しますと、ラインナンバーの後に自動的に：(コロン)が入り、INPUTやLETなどの命令の後が自動的に１桁あけられます。

10：INPUT␣R

自動的に１桁あきます。

自動的に：が入ります。

このことにより、計算機が、最初の数字をラインナンバー、アルファベットを１つの命令（INPUTという命令)として受け取り(理解し)、覚えたことを示します。

◎ **プログラムの編集**

プログラムを作成する場合、一度で正しくは、なかなか書けないものです。キー操作のミスや勘違いでまちがえることもあります。このような場合の処理方法について次に説明します。

○表示されているライン内の訂正や追加、削除は13ページのマニュアル計算での説明と同じように [◄] または [►] キーでカーソルを移動させ、[SHIFT] [DEL] や [SHIFT] [INS] を使っておこないます。

（例）　２０ＬＥＴ　Ｓ＝π＊Ｒ＊Ｒ　を
２０ ＩＮＰＵＴ　Ｓ＝π＊Ｒ＊Ｒ [ENTER] と押した場合の訂正

[►] または [◄] を操作　２０　ＩＮＰＵＴ　Ｓ＝π＊Ｒ＊Ｒ
↑ここを点滅させます。

[SHIFT] [DEL] を操作　２０　Ｓ＝π＊Ｒ＊Ｒ
↑ここが点滅します。

[SHIFT] [INS] を操作　２０　□□□ Ｓ＝π＊Ｒ＊Ｒ
↑必要な桁数分をあけます。

ＬＥＴ [ENTER]　２０：ＬＥＴ　Ｓ＝π＊Ｒ＊Ｒ

○ラインの追加は次のようにおこないます。

（例）　**プログラム1を計算機に入れるとき、次のように　ＬＥＴ　Ｓ＝π＊Ｒ＊Ｒ　をぬかしたとき。**

```
１０　ＩＮＰＵＴ　Ｒ
２０　ＰＲＩＮＴ　Ｓ
３０　ＥＮＤ
```

←この間に　ＬＥＴ　Ｓ＝π＊Ｒ＊Ｒ　を追加したい。

10ラインと20ラインの間に追加する場合は、この間のラインナンバー(11～19のいずれか)を使ってプログラムを書きます。

たとえば 15ＬＥＴ Ｓ＝π＊Ｒ＊Ｒ [ENTER] とします。そうすればプログラムは次のように正しくなります。

```
１０　ＩＮＰＵＴ　Ｒ
１５　ＬＥＴ　Ｓ＝π＊Ｒ＊Ｒ
２０　ＰＲＩＮＴ　Ｓ
３０　ＥＮＤ
```

注）プログラムは若いラインナンバーから書く必要はなく、計算機が自動的に若いラインナンバーから順に並べかえてくれます。

注）ラインナンバーだけを変えたいときは [◄] キーでラインナンバーを点滅させて、そのナンバーを変更します。

（例）３０　ＥＮＤ　のラインナンバーを４０に変えるとき

[◄] を操作　３０　ＥＮＤ
↑点滅させます。

４ [ENTER]　４０：ＥＮＤ　40ラインが書き込まれます。

３０ [ENTER]　＞　30ラインを削除します。

○あるラインをすべて削除するときはラインナンバー [ENTER] と押します。

＝参　考＝

ＮＥＷ式

式の値－１のアドレスまでのプログラム・データエリアを機械語エリアとして、それ以降のアドレスのプログラム・データエリアおよびすべての変数などをクリアします。

○プログラムの内容の確認は次の方法でおこなってください。

① [↓] キー
このキーを1回押すごとに現在表示しているラインの次のラインを表示します。また、このキーを押し続けますと、順次、次のラインを表示します。

② [↑] キー
このキーを1回押すごとに現在表示しているラインの前のラインを表示します。また、このキーを押し続けますと、順次前のラインを表示します。

③ LIST命令(リスト)
PROモードで LIST [ENTER] と押せば、もっとも若いラインを表示します。

LIST ＿＿＿＿＿ [ENTER]
ラインナンバー

と押せば、指示されたラインを表示します。(もし、指示されたナンバーのラインがない場合は、それよりも大きく、かつ一番近いラインが表示されます。)

●LIST命令はPROモードでのみ有効です。

注) 表示部には最大26桁表示することができますが、1ラインに長いプログラムを書きますと、その一部は表示できなくなります。このような隠れた部分を見るときは [►] キーを押し続けてください。そうすればカーソルが右端まで行き、そののち表示が左に送られて隠れている部分が出てきます。

3−3) プログラムの実行(RUN)

○プログラムの実行はRUNモードでおこないますので、[MODE] キーを押して表示部にRUNシンボルを表示させてください。
○続いて次の操作により実行を開始します。

RUN [ENTER]		(最も若いラインナンバーより実行)
RUN ＿＿＿＿＿ (ラインナンバー)	[ENTER]	(指定したラインナンバーより実行)

(例) プログラム1(21ページ)を実行しましょう。まずプログラムを計算機に入れてから次の計算をしてください。
①半径が5cmの場合
②半径が10cmの場合

[MODE]	>	RUN (RUNシンボルを表示させます。)
① RUN [ENTER]	?	
5	5_	
[ENTER]		78.53981634
[ENTER]	>	

② RUN [ENTER]　　?
10　　10_
[ENTER]　　314.1592654
[ENTER]　　>

○上例のようにＩＮＰＵＴでは？を表示して実行をいったん停止しますので、データを入力して [ENTER] キーで実行を再開させます。

○ＩＮＰＵＴ命令によりプログラム実行が停止しているとき、データを入力せずに [ENTER] キーを押した場合は、そのＩＮＰＵＴ命令が書かれているラインのＩＮＰＵＴ命令以降の命令を無視(スキップ)し、次のラインへ実行が移ります。

○ＩＮＰＵＴ命令において、文字変数に文字を入れるときは“　”で囲む必要はありません。また、文字変数の内容を消去するときは、ＩＮＰＵＴ命令によりデータの入力が求められているときに、”だけを入力して [ENTER] キーを押してください。(このとき表示部上側のシンボル以外は、なにも表示されません。)

(注) ＲＵＮ命令でプログラムを実行したとき、それまで確保されていた２文字の変数は消されます。(41ページを参照)

＝エラー表示と処理方法＝

プログラムを実行しているときにエラーが発生しますと、

ERROR 39 IN 30

のように、エラー番号と、エラーが発生したラインナンバーを表示します。
このときは、次のようにしてエラー処理をおこなってください。

① [CL] キーでエラーを解除してください。
(エラーが表示されているとき、あるいは [CL] でエラーを解除したとき、[↑] キーを押せば、押している間だけエラーが発生したプログラムラインを表示します。)

② [MODE] キーを押してＰＲＯモードにしてください。

③ [↓] キーあるいは [↑] キーを押してください。エラーが発生したプログラムラインが表示され、エラーが発生した位置がカーソルで示されています。

③ エラーが発生した原因を探して、13ページおよび、24 ページに記載の方法で訂正などをおこなってください。

●エラーの内容については167～169ページを参照してください。

３－４）いろいろな入力形（ＩＮＰＵＴ）

３－３項でプログラムの実行をおこないましたが、入力・出力が味気ない感じがしたでしょう。
この項と次の項では入力や出力に変化を持たせる入力形・出力形の基本を説明します。

〔例　題〕
半径入力時に　ＨＡＮＫＥＩ␣Ｒ＝　と表示させるような円の面積を求めるプログラムを作成してください。

〔解答例〕

```
10  INPUT "HANKEI␣R=", R
20  S=π*R*R
30  PRINT  S
40  END
```

本書ではスペースを意味しています。計算機に入れるときは SPACE キーを操作してください。

○計算機に書き込んで実際に実行し、表示を確認してください。また、10ラインを次のように変えて実行してみてください。

```
10  INPUT "HANKEI␣R=";R
```

この部分を,(コンマ)から;(セミコロン)に変えます。

〔説　明〕

◎　INPUT時にメッセージを表示させる方法です。

①INPUT "文字", 変数

（例）INPUT "A=" ,A
　　INPUT "A=", A, "B=", B

②INPUT "文字"; 変数

（例）INPUT "A=" ; A
　　INPUT "A=" ; A, "B=" ; B

①の形ではINPUT状態で数値を入力すればメッセージが消えますが、②の形ではメッセージは消えずに、入力したデータがメッセージに続いて表示されます。

（注）INPUT命令実行時、データの入力誤りなどでエラーになった場合は CL キーでエラーを解除すれば、データ入力待ちの状態にもどりますので、そのままデータの入力などをおこない、計算を続けることができます。

（注）51ページで説明しますGCURSOR命令で151以上が指定されているとき、続いてINPUT命令がある場合は、入力された内容が表示できませんのでご注意ください。

3－5）いろいろな出力形

〔例　題〕

①3－4項での例題において、円の面積を表示するとき、左半分に半径、右半分に面積を表示させるプログラムにしてください。

②また、円の面積を表示するときに　S=　と表示させ、それに続いて面積を表示させるプログラムにしてください。

〔解答例〕

①
```
10  INPUT "HANKEI␣R=", R
20  S=π*R*R
30  PRINT  R, S
40  END
```

② 10 INPUT "HANKEI␣R=",R
20 S=π*R*R
30 PRINT "S=";S
40 END

○プログラムを実際に実行してみてください。また、30ラインを次のように変えて実行し、その違いをみてください。

① 30 PRINT R;S ―― ,(コンマ)を;(セミコロン)に変えます。

② 30 PRINT "S=",S ―― ;を,に変えます。

〔説 明〕

◎ PRINTは基本的に次の使いかたができます。

①PRINT式

この形では式の値を表示部の右端を使って16桁以内で表示します。

注）式とは、数値や数値変数およびこれらを含んだ式をさします。

（例） 3，A，3＊4，3＊A＋5など

（例） PRINT S　　Sの内容を右端16桁以内に表示

②PRINT文字変数

PRINT“文字”

この形では文字あるいは文字変数の内容を表示部の左端から表示します。

（例） PRINT A$　　A$の内容を左端より表示

PRINT "MENSEKI␣S="　　文字を左端より表示

③PRINT { 式 / “文字” / 文字変数 } , { 式 / “文字” / 文字変数 }

26桁の表示部を左右13桁に分けて、,の左右の内容を表示させます。このときも、13桁の範囲内で数値は右ヅメ、文字は左ヅメにします。なお、表示内容が13桁を超える場合は、頭から13桁のみを表示します。

（例） PRINT A，S　　Aの値を左13桁内に、Sの値を右13桁内に右ヅメで表示。

PRINT "S=",S　　S＝を左13桁内に左ヅメで、Sの値を右13桁内に右ヅメで表示。

④PRINT { 式 / “文　字” / 文字変数 } ; { 数値変数 / “文　字” / 文字変数 } ;…; { 数値変数 / “文　字” / 文字変数 }

この形では文字や数値を表示部の左端よりツメて26桁の範囲内で表示します。

（例） PRINT "S=";S　　左端にS＝を表示し、続いてSの内容を表示します。

4．プログラム(その2)

プログラム(その1)では、プログラムの一番基本となる入力、計算、出力の3点について説明しましたが、ここではそれに加えてジャンプや判断、くり返し計算などを説明します。プログラムの大部分はこれらジャンプ、判断、くり返しなどでなっています。いわばプログラムの本当の基本となるものです。十分理解するようにしてください。

4－1）ジャンプと判断（GOTO，IF…THEN）

(1) ジャンプ

〔例　題〕

劇場に入場する人数を入力して、その料金を求めるプログラムを作ってください。
ただし、料金は1人 1,400円として、表示は左側に人数、右側に料金を表示させてください。

```
NINZUU    8     11200 EN
```

〔解答例〕(プログラム2)

```
10:INPUT "NINZUU=
   ";A
20:B=1400*A
30:PRINT "NINZUU␣
   ␣";A;"␣␣␣";B;"
   ␣EN"
40:GOTO 10
```

(注) このプログラムは10ラインから40ラインまでをくり返し実行するようになっていますので、プログラムの実行が終了しません。このようなプログラムにおいて実行を終了させるときは BREAK ON CL と押してください。

〔説　明〕

解答例のプログラムは、40ラインを除けばこれまで説明しましたプログラムと同じ形です。
ここでは40ラインに書かれていますGOTO命令について説明します。

◎ GOTO命令(ゴートゥー)

プログラムの実行は若いラインナンバーから順次実行されますが、このGOTO命令を実行すると、GOTOに続いて書かれているラインへジャンプして、ジャンプしたラインから実行します。

（例） GOTO　40　　ラインナンバー40へジャンプ
GOTO　5＊8　ラインナンバー40へジャンプ
GOTO　A　　Aの内容のラインへジャンプ(A＝40ならば40ラインへジャンプ)
GOTO "A"　　"A" ラベルのついているラインへジャンプ （次ページ参照）

（例）
```
      ┌ラベル
10:"A":INPUT "NIN
   ZUU=";A
20:B=1400*A
30:PRINT "NINZUU␣
   ␣";A;"␣␣␣";B;"
   ␣EN"
40:GOTO "A"
```

40ラインから10ラインへジャンプ
このようなジャンプをラベルジャンプといいます。

ラベル：ジャンプ先につけた名前のことで、ＧＯＴＯ命令などでラベルジャンプが指定されていますと、実行の際、そのラベルを探してそのラインにジャンプします。ラベルジャンプの場合、プログラムの訂正などでラインナンバーをつけかえても、プログラムは変更する必要がないので便利です。なお、同じラベルが２個以上書かれているときは、ラインナンバーの小さいほうへジャンプします。

(2) 判断

〔例　題〕

(1)項の例題において、人数が10人未満のときは1人あたりの料金を1,400円に、10人以上のときは団体料金として１人あたりの料金を1,200円にして料金を求めるプログラムを作ってください。

〔解答例１〕

```
10:INPUT "NINZUU=
   ";A
20:IF A>=10THEN 4
   0
30:B=1400*A:GOTO
   50
40:B=1200*A
50:PRINT "NINZUU␣
   ␣";A;"␣␣␣";B;"
   ␣EN"
60:GOTO 10
```

← 人数が10人以上かどうか判断
（10人以上のときは40ラインへジャンプ）

〔解答例２〕

```
10:INPUT "NINZUU=
   ";A
20:IF A>=10LET B=
   1200*A:GOTO 40
30:B=1400*A
40:PRINT "NINZUU␣
   ␣";A;"␣␣␣";B;"
   ␣EN"
50:GOTO 10
```

〔説　明〕

◎　ＩＦ命令(イフ)

ＩＦ命令はいろいろな判断をおこなう命令です。たとえば

ＩＦ　Ａ＞＝10

の例では、Ａの値が10以上(ＩＦ文が成立)ならばＩＦ文の後に続く内容を実行し、10未満、(ＩＦ文が不成立)ならば次のラインを実行します。

ＩＦ文に続く、Ａ＞＝10などの式を条件式といいます。条件式は通常次のような形で書きます。

ＩＦ　○○＝××　　等しいかどうか判断（○○は××に等しいか？）
ＩＦ　○○＞××　　大きいかどうか判断（○○は××より大きいか？）
ＩＦ　○○＞＝××　　以上かどうか判断（○○は××以上か？）
ＩＦ　○○＜××　　小さいかどうか判断（○○は××より小さいか？）
ＩＦ　○○＜＝××　　以下かどうか判断（○○は××以下か？）
ＩＦ　○○＜＞××　　等しくないかどうか判断（○○と××は等しくないか？）

（注）○○および××は式、変数を表わします。（５＊４、Ａ、８など）

ＩＦ文の後は通常ＴＨＥＮ60のように使い、指定したライン（60ライン）へのジャンプを指示します。（ＴＨＥＮ：ゼン）なお、解答例２のようにＩＦ文の後にステートメントを書いたほうがわかりやすい場合があります。（ＩＦ文の後に続いて、代入文を書く場合は必ずＬＥＴが必要です。）

条件式を通常の式にすることも可能です。たとえば

```
ＩＦ　Ａ＞Ｂ＋４　ＴＨＥＮ　６０
```

とした場合、Ａの値がＢ＋４の式の値よりも大きいならば60ラインを実行し、Ａの値がＢ＋４の式の値よりも大きくないならば、次のラインを実行します。

＝参　考＝

論理演算子（＝、＞、＜、＞＝、＜＝、＜＞）を用いて構成された式（論理式）は、それぞれ次の値をとります。

x、yは数値を示します。

＝※	$x=y$のとき：1 $x\neq y$のとき：0	＞＝	$x\geqq y$のとき：1 $x<y$のとき：0
＞	$x>y$のとき：1 $x\leqq y$のとき：0	＜＝	$x\leqq y$のとき：1 $x>y$のとき：0
＜	$x<y$のとき：1 $x\geqq y$のとき：0	＜＞	$x\neq y$のとき：1（＜＞は≠と同意味） $x=y$のとき：0

※ ＝を用いる論理式を例えばＡ＝数値、Ｂ＝式　のような形にしますと、計算機は論理式と判断せず、変数への入力（代入文）と判断しますので、このような場合は数値＝Ａ、式＝Ｂ　のような形にしてください。ただし、ＩＦ文における条件式の場合はこの限りではありません。

＝ご注意＝

本機は仮数部が最大10桁までの計算をおこなうことができますが、内部では計算の精度を上げるために仮数部12桁まで計算をおこない、11桁目を四捨五入処理したうえで計算結果として表示します。例えば5／9および5／9＊9の計算は内部で次のように計算されます。

```
5／9   →  5.55555555555E-01
                  └11桁目を四捨五入
表示   →  5.555555556E-01

5／9＊9→  4.99999999999E 00
                  └11桁目を四捨五入
表示   →            5
```

このように内部では、あくまで仮数部12桁の数値を用いて計算がおこなわれています。
したがって、計算を連続しておこなった場合と、区切って計算した場合とでは、その結果に差が生じることがあります。

（例1） $3^2-9=$

〈連続した計算〉

3 [SHIFT] [∧] 2 [−] 9 [ENTER] → −9E−11

〈区切った計算〉

3 [SHIFT] [∧] 2 [ENTER] → 9
[−] 9 [ENTER] → 0

また、IF文においても、条件式が連続的な計算では、この差のためにプログラムが目的通りに働らかない場合も生じますのでご注意ください。

（例2）

```
10  INPUT  A
20  IF  A↑2>=9 THEN  50
    ⋮
```

Aが3のとき、3↑2が計算機内部では8.99999999991E 00 となるためIF文が成立しません。

このようなときは、次のように変数を用いて〈区切った計算〉になるようにプログラムしてください。

```
10  INPUT  A
15  B=A↑2
20  IF  B>=9  THEN  50
    ⋮
```

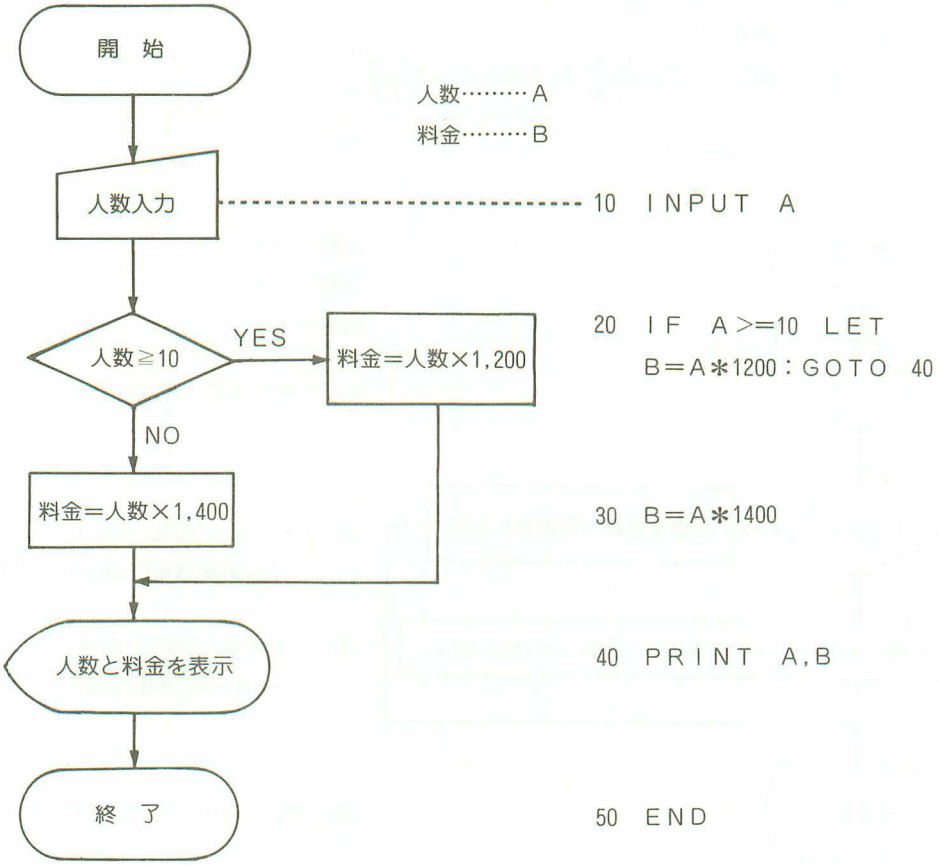
A↑2の結果を変数に代入し、その変数を用いて条件式を構成します。

べき乗計算は、log x と 10^x 計算を用いて計算されるため、計算機内との差が出やすくなります。
A↑B → $10^{B\log A}$ により計算

＝プログラムの考えかた＝

問題を見ていきなりプログラムを作ることは容易なことではありません。手順として計算順序の概略を図式化し、解析するのが普通であり、これを流れ図とかフローチャートといいます。フローチャートができあがればプログラムはほぼできたといってもよいほどです。プログラム上達のためには、まずフローチャートに慣れるのが早道ではないでしょうか。

フローチャートの次に各データをどの変数に割当てるか決めます。そして次にフローチャートの各部分をＢＡＳＩＣ命令におきかえていき、プログラムを作成します。



解答例ではいろいろな文字表示など、使いやすく、見やすくするためにつけていますが、これらは一種のアクセサリーと考えてもよいでしょう。まず上のようなフローチャートを作り、上のようなプログラムを作ることが第一です。慣れるにしたがって、いろいろなアクセサリーに工夫をこらせばよいでしょう。

〔例　題〕

金額と日数を入力して利息を求めるプログラムを作ってください。

ただし、日数が30日以下は日歩２銭、30日を越える分については日歩４銭とします。

〔解答例〕

●解析

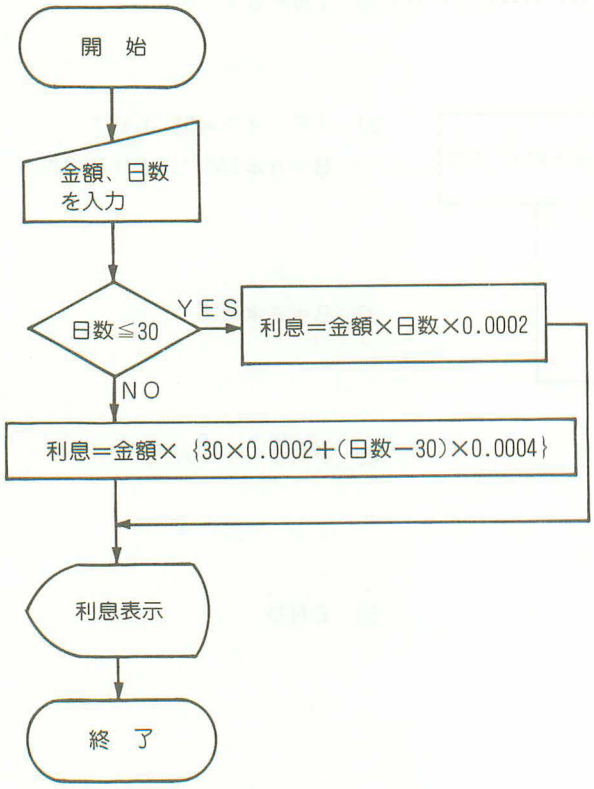
日数が30日以下のとき

　利息＝金額×日数×$\frac{0.02}{100}$

日数が30日を越えたとき

　利息＝金額×$\left\{30日\times\frac{0.02}{100}+(日数-30)\times\frac{0.04}{100}\right\}$

●フローチャートとプログラム




```
金額……A
日数……B
利息……C

10  INPUT  A,B

20  IF  B<=30  LET
      C=A*B*0.0002：GOTO40

30  C=A*(30*0.0002+
      (B-30)*0.0004)

40  PRINT "RISOKU ";C

50  END
```

4−2）くり返し計算（FOR…TO…STEP，NEXT）

〔例　題〕
sin0°からsin90°まで5度きざみで計算し、その結果を表示させるプログラムを作ってください。

〔解答例〕

```
10:FOR A=0TO 90
   STEP 5
20:B=SIN A
30:PRINT "SIN";A;
   "=␣";B
40:NEXT A
50:END
```

FOR−NEXT ループ

（RUN ENTER で実行を開始したあとは、ENTER キーでつぎつぎに実行させます。）

〔説　明〕

◎ **FOR……NEXT文(フォー、ネクスト)**

FOR　NEXT文の一般形は次のようになります。

FOR　数値変数＝初期値　TO最終値　STEPきざみ
　　⟆
NEXT　数値変数

数値変数が初期値から始まって、指定されたきざみずつ増加していき、最終値以上になるまでFORとNEXTの間の計算をくり返し実行します。(このくり返し部分をFOR−NEXTループと呼びます。)

○　FORとNEXTは必ず対にして使い、FORの後の数値変数とNEXTの後の数値変数は同一の変数でなければなりません。

○　きざみが1のときはSTEP1は省略できます。

○　初期値、最終値、きざみを負の値にすることもできます。
（例）10　FOR　A＝0　TO　−90　STEP　−5

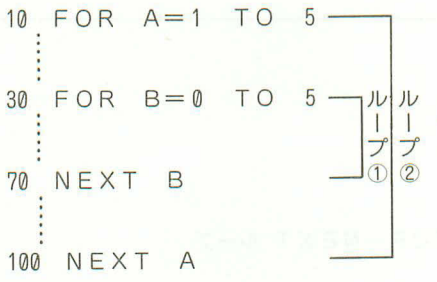
◎ **FOR−NEXTループの注意事項**

FOR−NEXTループを使う場合に注意すべきことがいくつかありますので次に説明します。

①　最終値ときざみ(ステップ数)は次の範囲内でなければなりません。
　　−32768〜32767　　（ただし、きざみ≠0）
きざみの値に小数部が含まれる場合、小数部は無視されます。

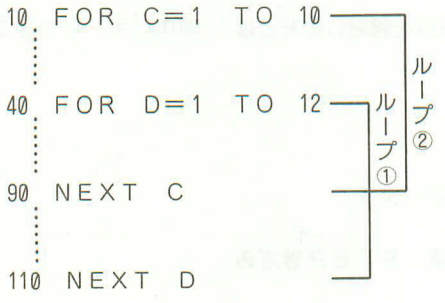
② ＦＯＲ－ＮＥＸＴループの中に、別のＦＯＲ－ＮＥＸＴループを入れることができます。ただし、中に入るＦＯＲ－ＮＥＸＴループは外のＦＯＲ－ＮＥＸＴループ内に完全に入っていなければなりません。

（例１）



ループ①がループ②の中に完全に入っていますので、この使い方は正しい。

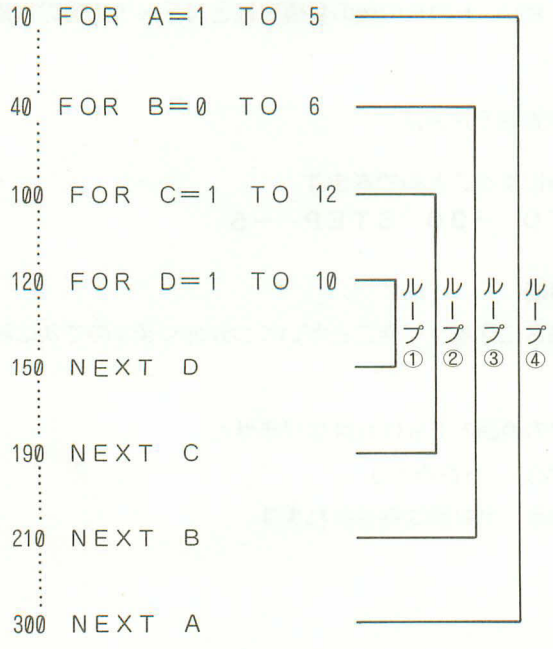
（例２）



ループ①とループ②が交差していますので
この使い方は誤り　（ＥＲＲＯＲ　２）

なお、ＦＯＲ－ＮＥＸＴループは３段、４段……と重ねて使う（深みをもたせる）ことができますが、いずれも、中に入るループは外のループ内に完全に入っていなければなりません。

（例）



ＦＯＲ－ＮＥＸＴループは
重ねて使うことができます。
（段数については41ページを参照）

③ FOR−NEXTループ内から外に飛び出すことはできますが、外からFOR−NEXTループ内に飛び込むことはできません。

（例１）

```
30  FOR  A=1  TO  20
60  IF  C>D  THEN  100
90  NEXT  A
100  C=D*2
```

飛び出し

ループ内から飛び出すことはできます。

（例２）

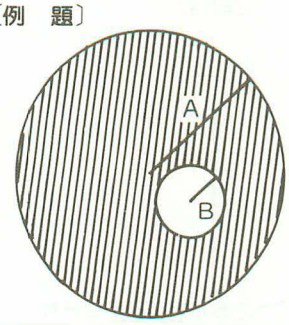
```
40  GOTO  80
60  FOR  H=1  TO  10
80  A=C*C+10
100  NEXT  H
```

飛び込み

ループの外からループ内に飛び込むことはできません。（ＥＲＲＯＲ　２）

4−3）サブルーチン（ＧＯＳＵＢ，ＲＥＴＵＲＮ）

〔例　題〕



図のような２円があるとき、２円の半径を入力して斜線部の面積を求めるプログラムを作ってください。

〔解答例〕

```
10:INPUT "A=";A,"
   B=";B
20:R=A:GOSUB 70
30:C=S:R=B:GOSUB
   70
40:D=S
50:PRINT C-D
60:END
70:S=π*R*R
80:RETURN
```

メインルーチン

サブルーチン

〈使用メモリー〉

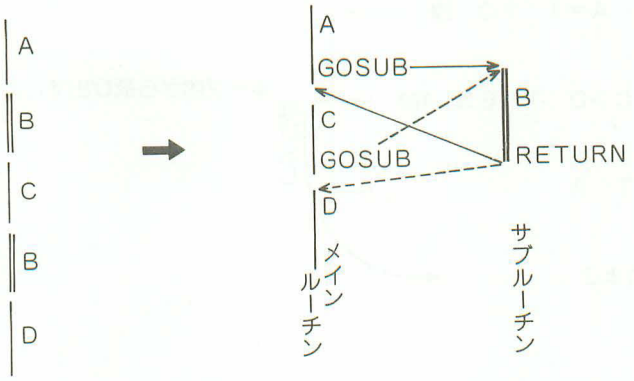
A　大円の半径
B　小円の半径
C　大円の面積
D　小円の面積

サブルーチン

R　円の半径
S　円の面積

〔説　明〕

○何度も同じような処理をおこなうときに、その処理をサブルーチン化すると便利になります。この例題はほんの一例ですが、共通処理部をたびたび使ったり、共通処理部が長い場合には全体のプログラムを短縮でき、またプログラムが見やすく、それだけ誤りも少なくなります。



上の左図のように共通部(Ｂ)があるとき、その部分をサブルーチンにすると右のようになります。サブルーチンへのジャンプはサブルーチンのおかれているラインナンバー(またはラベル)をＧＯＳＵＢの後に続いて書いて指示します。またサブルーチンはＲＥＴＵＲＮ命令でメインルーチン(ＧＯＳＵＢのあった元のプログラムをサブルーチンに対しメインルーチンと呼びます。)のＧＯＳＵＢのあった次の命令を引続き実行します。

●サブルーチンの中にサブルーチンを入れることもできます。

●ＧＯＳＵＢ命令の一般形は次のようになります。

ＧＯＳＵＢ式

あるいは

ＧＯＳＵＢ“文字”

ＧＯＳＵＢ命令の指定の形はＧＯＴＯ命令と同様です。

4－4）定義付けプログラム

複数のプログラムが書き込まれているときに、2番目以降のプログラムを実行するときや、プログラムの途中から実行するときには、通常次の方法でスタートさせます。

① RUN　ラインナンバー [ENTER]
② GOTO　ラインナンバー [ENTER]

この方法により、指定したラインからプログラムの実行が開始されます。この方法では操作に手間がかかり、また、ラインナンバーを常に知っていなければならず、なかなかわずらわしいものです。このようなとき、プログラムラインの始めに "A" や "B" のようにラベルを書いておくことにより、[DEF] [A] や [DEF] [B] と押すだけでプログラムの実行を開始させる方法があります。

〔例　題〕

次の3つの計算をおこなうプログラムを作ってください。このとき、①のプログラムは [A] キーに、②のプログラムは [S] キーに、③のプログラムは [D] キーにそれぞれ定義付けをおこなって、[DEF] [A] 、[DEF] [S] 、[DEF] [D] と押すことで、それぞれのプログラムを実行できるようにしてください。

① $c=\sqrt{a^2+b^2}$　ピタゴラスの定理
② $V=\frac{4}{3}\pi r^3$　球の体積
③ $c=\sqrt{a^2+b^2-2ab\cos\theta}$　余弦定理

〔解答例〕

```
        ←ラベルA
10:"A":INPUT "A="
   ;A,"B=";B
20:C=√(A*A+B*B)          ①
30:PRINT "C=";C
40:END
        ←ラベルS
50:"S":INPUT "R="
   ;R
60:V=4/3*π*R^3           ②
70:PRINT "V=";V
80:END
        ←ラベルD
90:"D":INPUT "A="
   ;A,"B=";B,"Z="
   ;Z
100:C=√(A*A+B*B-2*
   A*B*COS Z)            ③
110:PRINT "C=";C
120:END
```

〔説　明〕

◎ **ラベル**

ラベルは飛び越しの行先につけた名前(名札)のことです。前例のように [A]、[S]、[D] などのキーに対応する文字・記号を“　”で囲ってラインナンバーのすぐ後に書き込んでおけば、[DEF] [A]、[DEF] [S]、……と押したときに、そのキーに対応するラベルを探して、そのラベルの書かれているラインからプログラムが実行されます。

◎ **定義付けキー**

定義付けのできるキーは次の18キーです。

[A]、[S]、[D]、[F]、[G]、[H]、[J]、[K]、[L]、
[Z]、[X]、[C]、[V]、[B]、[N]、[M]、および [=]、[SPACE]

注）定義付けキーによりプログラムの実行開始を指定したとき、そのラベルをつけた（定義付けされた）プログラムがない場合はエラー(ＥＲＲＯＲ 11)になります。

＝よく使う命令の簡単な入れかた＝

これまで、ＩＮＰＵＴやＰＲＩＮＴなどの命令はキーを１つずつ押して、すべての文字を入れていましたが、次のようなことを知っていれば、操作を簡略することができます。

① アルファベットのキーのうち、次のキーにはそれぞれ次の命令が割り当てられています。

[Q]	[W]	[E]	[R]	[T]	[Y]	[U]	[I]	[O]	[P]
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
INPUT	PRINT	USING	GOTO	GOSUB	RETURN	CSAVE※	CLOAD※	MERGE※	LIST

これらのキーを [DEF] キーに続いて押せば、割り当てられている命令が入力されます。

※ ＣＳＡＶＥ、ＣＬＯＡＤ、ＭＥＲＧＥの各命令はカセットテープレコーダー用の命令であり別売のカラーグラフィックプリンタ(カセットインターフェイス内蔵)を取りつけたときにのみ使用することができます。このため、カラーグラフィックプリンタを取りつけていないときは、これらの命令は入力できず、替りに〜シンボルが入力されます。

② 各命令には省略した形(省略形)で入力できるものがあります。たとえば、ＰＲＩＮＴはＰ.　ＧＯＳＵＢはＧＯＳ.　と入力すれば計算機はそれぞれの命令として理解することができます。これらの省略形は171 ページの関数・命令一覧に記載していますので参照してください。

＝スタックエリアについて＝

（忙しい人は、この項は飛ばして、次へ進んでください。）

本機はプログラムを実行する際、ＦＯＲ－ＮＥＸＴやＧＯＳＵＢ－ＲＥＴＵＲＮ命令を一時記憶しておくための、あるいは計算の優先順位やカッコにより保留される、計算命令やデータを一時記憶しておくためのスタックエリアを持っています。

プログラム実行中あるいは計算実行中に、すぐに実行してしまうことのできない命令や計算をこのスタックエリアに一時記憶させておいて、実行できるようになったときに呼び出して実行します。このスタックエリアの容量は196バイトあります。

次にＦＯＲ－ＮＥＸＴやＧＯＳＵＢ－ＲＥＴＵＲＮなどのスタックエリアでの使用バイト数を表示します。

命令データ	使用バイト数
ＦＯＲ－ＮＥＸＴ	12
ＧＯＳＵＢ－ＲＥＴＵＲＮ	6
データ	8
計算命令（＋－＊／など）	2

ＦＯＲ－ＮＥＸＴやＧＯＳＵＢ－ＲＥＴＵＲＮを何段も深みを持たせた場合や何重にもカッコを使用した計算をおこなった場合は使用バイト数が196バイトを越えてエラーになることがあります（ＥＲＲＯＲ　14、15）

＝プログラムの実行開始方法によるスタート状態の違いについて＝

（忙しい人はこの項を飛ばして、次へ進んでください。）

プログラムの実行開始方法には、ＲＵＮ命令によるもの、ＧＯＴＯ命令によるもの、 DEF A 、などの定義付けキーによるものがありますが、これらの開始方法により、変数のクリアや状態の解除などに違いがありますので次に示します。

ＲＵＮ	ＧＯＴＯ	定義付けキー
〈実行方法〉 ＲＵＮ　ENTER ＲＵＮ式　ENTER	〈実行方法〉 ＧＯＴＯ　ENTER ＧＯＴＯ式　ENTER	〈実行方法〉 DEF に続くＡ，Ｓ，Ｄ，Ｆ，Ｇ，Ｈ，……
○表示をクリア ○カーソルクリア ○ウェイト指定解除 ○表示フォーマット指定解除 ○配列変数および２文字変数クリア ○ＦＯＲ－ＮＥＸＴ、ＧＯＳＵＢスタックのクリア ○ＯＮ　ＥＲＲＯＲ解除 ○ＲＥＡＤ文に対するＤＡＴＡの初期化	○表示をクリア ○ＦＯＲ－ＮＥＸＴ、ＧＯＳＵＢスタックのクリア	○ＦＯＲ－ＮＥＸＴ、ＧＯＳＵＢスタックのクリア

5．変数

変数については2－3）でふれましたが、ここでは変数の構成などについて、さらにくわしく説明します。

5－1）変数の構成

- 変数
  - 数値変数
    - 数値固定変数（A～Z）
    - 数値単純変数（AA、CF、B2など）
    - 数値配列変数
  - 文字変数
    - 文字固定変数（A$～Z$）
    - 文字単純変数（AA$、B2$など）
    - 文字配列変数

数値配列変数、文字配列変数については、のちほど5－2）で説明します。

◎ **数値変数**

数値変数は仮数部10桁、指数部－99～99で表わすことのできる数値を扱います。

○数値固定変数

A～Zの変数

○数値単純変数

AA、AB、F2、H3など、2文字の変数

変数の名前には、英大文字と英大文字、あるいは英大文字と数字の組み合わせ(1文字目が必ず英大文字)を用いることができます。(3文字以上の名前をつけた場合は前2文字が有効となります。)

◎ **文字変数**

次の文字変数は最大16字までの文字・記号を扱います。

○文字固定変数

A$～Z$の変数

○文字単純変数

AA$、AB$、F2$、H3$など、変数名(2文字)の後に$記号をつけた変数

5－2）配列変数（DIM）

ある同数のデータをたくさん扱う場合がありますが、このようなときに配列変数を用いますと大変便利です。

〔例　題〕

INPUT文を使用して、任意の数値を10個読み込ませ、1番大きな数値を探し出すプログラムを作ってください。

```
10:DIM A(9)
20:FOR B=0TO 9
30:INPUT "DATA=";
   A(B)
40:NEXT B
50:M=A(0)
60:FOR B=1TO 9
70:IF M>=A(B)THEN
   90
80:M=A(B)
90:NEXT B
100:PRINT "SAIDAI␣
   WA␣";M
110:END
```

FOR－NEXTループ
データの読み込み

FOR－NEXTループ
データの比較

〔説　明〕

◎ 配列変数を使うときは、事前にその変数として使う場所をメモリーの中に確保する必要があります。配列変数の場所を確保するためにはＤＩＭ(ディメンジョン)命令を使って、その変数名と大きさを定義します。たとえばＤＩＭ　Ａ(９)は配列変数Ａ(　)のためにＡ(０)からＡ(９)までの10個の変数がメモリー内に確保されることを意味します。（下図参照）

A(0)	A(1)	A(2)	A(3)	A(4)	A(5)
A(6)	A(7)	A(8)	A(9)		

◎ Ａ(０)、Ａ(１)…のように書かれている変数の、カッコの中の数字を添字といいます。
本機では配列変数に添字が１個のものと２個のものを使用することができ、それぞれを一次元配列、二次元配列といいます。

一次元配列

ＤＩＭ　Ａ(５)　　Ａ(０)、Ａ(１)……Ａ(５)の６個の変数を確保

二次元配列

ＤＩＭ　Ａ(３，４)　　Ａ(０，０)、Ａ(０，１)、Ａ(０，２)……Ａ(３，４)の20個の変数を確保（下図参照）

A(0,0)	A(0,1)	A(0,2)	A(0,3)	A(0,4)
A(1,0)	A(1,1)	A(1,2)	A(1,3)	A(1,4)
A(2,0)	A(2,1)	A(2,2)	A(2,3)	A(2,4)
A(3,0)	A(3,1)	A(3,2)	A(3,3)	A(3,4)

◎ **文字配列**

配列変数にも文字変数があり、固定変数や単純変数と同様に＄記号をつけて指定します。

（例） ＤＩＭ　Ａ＄(９)　　**Ａ＄(０)、Ａ＄(１)……Ａ＄(９)の10個の変数を確保**

ＤＩＭ　Ａ＄(５，４)　　**Ａ＄(０，０)、Ａ＄(０，１)……Ａ＄(５，４)の30個の変数を確保。**

◎ **文字配列の拡張**

文字固定変数や文字単純変数は、１個の変数に最大16文字まで記憶できる固定長の変数でしたが、配列変数では最大80文字以内で、任意に変数の長さを指定することができます。

（例） ＤＩＭ　Ａ＄(９)＊３０　　**Ａ＄(０)、Ａ＄(１).…Ａ＄(９)の変数にはそれぞれ最大30文字まで格納することができます。**

ＤＩＭ　Ａ＄(５，４)＊６　　**Ａ＄(０，０)、Ａ＄(０，１)……Ａ＄(５，４)の変数にはそれぞれ最大６文字まで格納することができます。**

上例のように、ＤＩＭ文に ＊式 をつけて変数の長さを指定します。指定できる文字数は１～80文字で、指定しない(＊式 がない)ときは自動的に16文字に設定されます。

◎ 配列変数はＡ(Ｂ)、Ｂ＄(Ｃ)、Ａ1(Ｃ，Ｄ)のように、添字を変数や式で指定することもできます。

◎ 添字は理論的には０～255までの数値を用いることができます。
ただし、変数の長さ、プログラムの大きさなどにより、添字で指定した数だけ、変数が確保できない場合があります。

◎ 配列変数をたくさん定義する場合は次のような形で一度に定義できます。

ＤＩＭ　Ａ(２０)，Ｂ(５，７)，Ｃ＄(１０，３)＊６

注）同一プログラム内で、同じ変数名の配列を再定義することはできません。

ＤＩＭ　Ａ(２０) とＤＩＭ　Ａ(３，４)は同じＡ(　)という変数名になります。

＝参　考＝

固定変数Ａ～Ｚ、Ａ＄～Ｚ＄も添字つきの変数として指定することができます。

@(１)～@(26)……数値変数Ａ～Ｚを添字つきの変数として指定

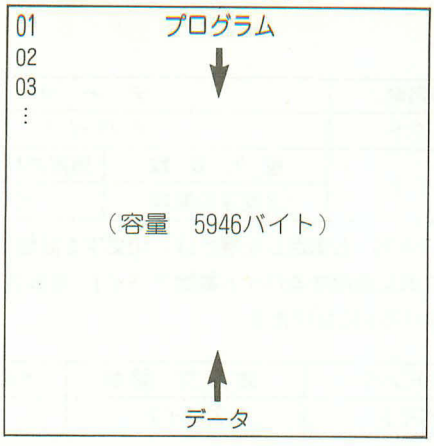
@＄(１)～@＄(26)……文字変数Ａ＄～Ｚ＄を添字つきの変数として指定

5－３）メモリーの構成と変数

変数については2－３）で少しふれましたが、ここでは本機の持つメモリーの構成と、変数についてさらにくわしく説明します。

◎ **メモリーの構成**

〈プログラム・データエリア〉



プログラムがこちら側から書き込まれていきます。

ＡＡやＡ１＄のような2文字の変数、Ａ(１)やＢ(５，８)のような配列変数はこちら側から確保されます。

〈データ専用エリア〉（容量 624バイト）

Ａあるいは@(１)	
Ｂあるいは@(２)	
Ｃあるいは@(３)	
⋮	
Ｚあるいは@(26)	
Ａ＄あるいは@＄(１)	
Ｂ＄あるいは@＄(２)	
Ｃ＄あるいは@＄(３)	
⋮	
Ｚ＄あるいは@＄(26)	

数値固定変数（26メモリー）

文字固定変数（26メモリー）

〈リザーブメモリーエリア〉

（容量　188バイト）

後ほど説明しますリザーブ内容、キーシンボルを記憶します。

〈拡張メモリー：別売〉

メモリーは上記のような構成になっています。プログラムエリアはプログラムメモリー、データメモリー(変数)のいずれにも使用することのできるメモリーです。図で示すように、プログラムと変数はそれぞれ逆方向から書き込まれ、確保されていきます。これに対してデータ専用エリアは常にデータメモリー(変数)として確保されています。通常、データメモリー(変数)として使用できるのはこの2種類のメモリーエリアですが、プログラム・データエリアを使用するときは書き込まれているプログラムの長さによって変数として使える大きさが変ってきますので注意が必要です。

次に各変数を確保する場合に使用するバイト数およびプログラムの各命令などの占めるバイト数を記しておきますので参考にしてください。

変　　数	変数の名前	データ	
数値変数	7バイト	8バイト	
文字変数	7バイト	配列変数	指定されたバイト数※
		2文字の変数	16バイト

※たとえば　DIM　A$(2，3)*10　と指定した場合は、10文字を記憶できる変数を12個確保しますが、これに使用するバイト数は7バイト(変数名)＋10バイト(文字数)×12個＝127バイトになります。

構成要素	ラインナンバー	命令文、関数	その他 ENTER
使用バイト数	3バイト	2バイト	1バイト

注) プログラム・データエリアはプログラムとデータ用として共用しますので、プログラムの入力により、このエリアがいっぱいになり、ついには、あらかじめDIM指定などにより確保されているデータ領域を侵してしまう場合があります。
このときはデータが消され、入力したプログラムが優先して書き込まれますが、データが消去されたことを示すため、エラー(ERROR　177～181)になります。

5-4) 変数(メモリー)の内容を消すには(CLEAR)

◎　**個々の変数をクリア(消去)するには次のようにします。**

(例)　A＝0
　　B(1)＝0
　　　} 数値変数は0を代入することによりクリアします。

　　A$＝"
　　BC$＝"
　　　} 文字変数は"　を代入することによりクリアします。

◎　**CLEAR命令(クリア)**

すべての変数を一度にクリアするには、CLEAR命令を実行します。CLEAR命令の一般形は次のようになります。

CLEAR

●この命令ではプログラムはクリアされず、保護されます。

6．表示に関するＢＡＳＩＣ

これまでの説明では、ＩＮＰＵＴ命令やＰＲＩＮＴ命令での表示のしかたに多くの変化はありませんでしたが、ここでは表示に多くの変化を持たせる指定のしかたや、自由な文字や記号の作りかたなどを説明します。

6－1）いろいろな表示のさせかた（PRINT,USING,CURSOR,WAIT,PAUSE,CLS）

〔例　題〕

整数ｎを入力し、その階乗を求めるプログラムを作ってください。

n!＝1×2×3×……×n

このとき、次のような表示をさせるようにしてください。

①プログラム実行中は　○　NO␣KAIJOU　と表示させてください。

　　↑入力した数値を表示させます。

②プログラム実行後、①の表示に続いて結果を小数部５桁の指数方式で表示させてください。

〔解答例〕

```
10:INPUT "DATA␣N=
   ";N
20:WAIT 60              ←ＰＲＩＮＴ命令での停止時間を指定しています。
30:PRINT USING "#
   ##";N;"NO␣KAIJ
   OU";
40:B=1
50:FOR A=1TO N          ┐
60:B=B*A                ｎの階乗を計算するループです。
70:NEXT A               ┘
80:WAIT                 ←ＰＲＩＮＴ命令での停止時間を無限に指定
90:PRINT "=";
   USING "#.#####
   ^";B                 ←表示の形(フォーマット)を指定
100:END
```

〔説　明〕

◎　**ＰＲＩＮＴ命令**

ＰＲＩＮＴ命令についてはすでに3－5）で説明しましたが、ここでもう少し説明します。

ＰＲＩＮＴ命令には3－5）で説明しました形のほかに次の形の指定ができます。

$$\text{PRINT}\left\{\begin{array}{c}\text{式}\\ \text{“文字”}\\ \text{文字変数}\end{array}\right\};$$

$$\text{PRINT}\left\{\begin{array}{c}\text{式}\\ \text{“文字”}\\ \text{文字変数}\end{array}\right\};\left\{\begin{array}{c}\text{数値変数}\\ \text{“文字”}\\ \text{文字変数}\end{array}\right\};\cdots\cdots;$$

この形の場合、ＰＲＩＮＴ命令実行により、文字や数値を表示部の左端より表示します。そして、次にＰＲＩＮＴ命令などの表示命令がありますと、その指定内容を前の表示に続いて表示します。

（例）
```
10:WAIT 10
20:FOR A=1TO 26
30:PRINT "#";
40:NEXT A
50:WAIT :PRINT
60:END
```

← ＦＯＲ－ＮＥＸＴループ
　表示部の左端から右端まで
　＃記号を表示させます。
← 表示されている内容を保持したまま
プログラムの実行を停止します。

◎ ＵＳＩＮＧ命令（ユージング）

ＵＳＩＮＧ命令は数値などを表示するときに、その表示フォーマットを指定します。

○ ＵＳＩＮＧ命令の一般形

① ＵＳＩＮＧ　"＃＃＃"　　符号を含めて整数３桁を表示

② ＵＳＩＮＧ　"＃＃＃．"　　符号を含めて整数３桁と小数点を表示

③ ＵＳＩＮＧ　"＃＃＃．＃＃"　　符号を含めて整数３桁と小数点と小数点以下２桁表示

④ ＵＳＩＮＧ　"＃＃．＃＃＾"　　小数以下２桁までの指数方式での表示

（このとき、仮数部の整数は符号を含めて２桁、指数部は符号を含めて４桁（Ｅ－ＯＯ）が自動的にとられます。）

⑤ ＵＳＩＮＧ　"＃＃＃，＃＃＃．"　　符号と３桁区切りマーク（，）を含めて整数６桁と小数点を表示

（数値を表示するときは　，　も１桁と数えますので、たとえば－1,234,567.を表示させるときは
　ＵＳＩＮＧ "＃＃＃＃＃＃＃，＃＃＃．"
のように＃マークを10個入れる必要があります。）

⑥ ＵＳＩＮＧ　"＋＃＃＃"　　正の数値に＋符号をつけます。
数値は整数２桁（符号を含めて３桁）を表示します。

⑦ ＵＳＩＮＧ　"＊＃＃＃＃．"　　符号を含めて整数５桁と小数点を表示しますが、数値の整数が指定桁より少ない場合は、少ない桁数だけ＊記号をつけます。
この指定で－１を表示させた場合は
－＊＊＊１．のように表示

⑧ ＵＳＩＮＧ　"＆＆＆＆＆＆"　　文字を６桁表示

（注） 整数部を指定する＊と＃の合計の数は次の範囲内でお使いください。

3桁区切りマーク（，）をつけない場合：11個以内
3桁区切りマーク（，）をつける場合　：14個以内

本機は数値の有効桁数が10桁になっています。ＵＳＩＮＧ命令により整数部が10桁を越えるフォーマットを指定して、ＰＲＩＮＴ命令やＬＰＲＩＮＴ命令（プリンタ用の命令）により、数値を表示あるいは印字させたときに整数部が10桁を越えていますと正しくない数値を表示（印字）する場合があります。

（例） ＲＵＮモード

```
USING"###############"[ENTER]→>
PRINT 888888888888 [ENTER]     →888888888800
(LPRINT 888888888888 [ENTER]   →888888888880)
       └── 12桁 ──┘
```

●ＬＰＲＩＮＴ命令については84ページを参照ください。

○ ＵＳＩＮＧ文を単独に使います。

```
10 A=12.345, B=-34.567, C=5
20 USING "###.##"
30 PRINT A; B; C
40 END
```

表示例

```
 12.34-34.56  5.00
```

␣スペース　␣スペース ␣スペース

○ ＰＲＩＮＴ文の中にＵＳＩＮＧ文を含め、ＰＲＩＮＴ　ＵＳＩＮＧ文として使います。

```
10 A=4 ,B=10 ,C=10 .7703
20 PRINT "A=" ;USING "###" ;A;
   "␣B=" ; B;"␣C=" ;
   USING "###.###" ;C
30 END
```

表示例

```
A=  4 B= 10 C= 10.770
```

␣␣スペース　␣スペース　␣スペース　␣スペース　␣スペース

○ ＵＳＩＮＧ指定の解除

ＵＳＩＮＧ

ＵＳＩＮＧで表示フォーマットの指定をおこないますと、以降に実行されるＰＲＩＮＴ

命令には、すべてそのフォーマット指定が有効になりますので、不必要なときは本命令によってフォーマット指定を解除しておきます。

（例）
```
10:A=34,B=7
20:C=A/B
30:USING "#####"
40:PRINT "A=";A
50:PRINT "B=";B
60:PRINT USING ;"          ←――USINGによりフォーマット指定解除
   C=";C
70:END
```

◎ WAIT命令（ウェイト）

WAIT命令は、PRINT命令によるプログラムの停止時間を指定する命令です。WAIT命令による指定はPRINT命令のすべての形に対して有効で、プログラムは指定された時間停止し、その後、自動的に実行を再開します。

○ 停止時間は次のように指定します。

WAIT 式

式の値により、決められている時間だけプログラム実行を停止して内容を表示し、その後プログラムの実行を再開します。

式の値は0～65535まで指定できます。なお、式の値1は約1/64秒に相当します。

○ WAIT指定の解除

WAIT

WAITでの指定は、以降のPRINT命令に対してすべて有効となるため、時間を無限にしたいときなどは、本命令によって時間指定を解除してください。

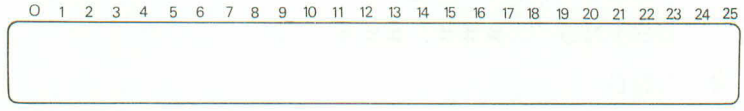
◎ CURSOR命令（カーソル）

CURSOR命令はPRINT命令などにより表示される内容の表示開始位置（ポジション）を指定します。

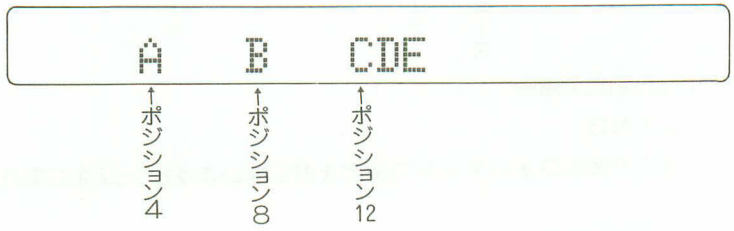
CURSOR 式　　式の値で表示開始ポジションを指定します。

CURSOR　　指定を解除します。

表示ポジションを下に示します。



（例）
```
10  CURSOR  4 : PRINT "A"
20  CURSOR  8 : PRINT "B"
30  CURSOR  12 : PRINT "CDE"
40  END
```



●CURSOR命令で表示開始位置を指定した場合は、以降のPRINT命令で表示された内容の前後は、それまで表示されていた内容が保持されます。この機能を使えば、表示の一部分だけを変えることもでき、応用範囲も広がります。なお、表示をクリアするときはCLS命令を用います。

◎ PAUSE命令（ポーズ）

PRINT命令が表示をおこなってプログラムの実行を停止、あるいはWAIT命令により停止時間が指定されるのに対して、このPAUSE命令は停止時間が一定（約0.85秒）に決っており、その時間だけ表示をおこなって次のプログラムに実行が移るところが異っているだけで、そのほかはPRINT命令と同じです。

◎ CLS命令（クリアーズ）

表示内容を消し、表示開始ポジションを0位置にもどします。

6－2）文字や記号の作りかた（GCURSOR, GPRINT, POINT）

本機で表示される文字や記号の1つ1つは小さい点(ドット)を使って作られています。
本機内には文字や記号を表示するドットパターンのデータがたくさん記憶されていますが、このデータによらず、自由な文字や記号を作ることができれば、計算機をもっといろいろなことに活用することができます。ここでは、そのような文字や記号の作りかたについて説明します。

〔例　題〕
表示部の中ほどに◆マークを3個表示させるプログラムを作ってください。

〔解答例1〕

```
10:GCURSOR &42          ←― &は16進数を示す記号です。
20:WAIT 0               ←― GPRINT命令での停止時間を0に指定
30:FOR A=1TO 3
40:GPRINT "00081C
   3E1C0800";           ←― ◆マークを3個作っています。
50:NEXT A
60:WAIT :PRINT          ←― 表示を保持して、実行を停止
70:END
```

〔解答例2〕

```
10:GCURSOR 66
20:WAIT 0
30:FOR A=1TO 3
40:GPRINT 0;8;28;
   62;28;8;0;
50:NEXT A
60:WAIT :PRINT
70:END
```

〔説　明〕

◎　GCURSOR命令(グラフィックカーソル)

GCURSOR命令は、基本的にはCURSOR命令と同じですが、CURSOR命令が桁の単位で表示開始ポジションを指定したのに対し、GCUSOR命令はドット(点)の列の単位で表示開始ポジションを指定します。

○　表示部はタテに7個並んだドットの列が156列あります。この156列に左側から順に0～155(&0～&9B)までの番号をつけ、これにより表示開始ポジションを指定します。

(例)　GCURSOR　5　　5番のポジション(左から6列目)を指定します。
GCURSOR　62　　62番のポジション(左から63列目)を指定します。
GCURSOR　&3E　16進数での3E番(10進数での62番)のポジションを指定します。

◎　16進数

16進の数値は、10進数の10～15に対する数字をA、B、C、D、E、Fと規定した数字(0～9ABCDEF)を用いて表わします。

10進数→…9　10　11　12　13　14　15　16　17　……30　31　32　……
16進数→…9　A　B　C　D　E　F　10　11　……1E　1F　20　……

プログラム上では10進数と16進数を区別するため、16進数値の前に&記号をつけて表わします。

(例)　&14　　16進数の14を表わします。
&2A　　16進数の2Aを表わします。

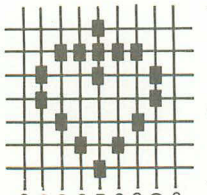
◎　GPRINT命令(グラフィックプリント)

GPRINT命令はドットパターンを指定することにより、そのパターンを表示する命令で次のような形で指定します。

GPRINT "○○○○○……"
それぞれタテ1列のドットパターンを指定します。

ドットの指定はタテ7ドットを上4ドット、下3ドットに分割し、それぞれを16進数字により指定します。

(例)　GPRINT "0C 12 22 47 22 12 0C"
上側のドットパターンを表わします。
下側のドットパターンを表わします。



0C22722C0 ←　上側のドットパターンを16進数字で表わした場合。
0012421 00 ←　下側のドットパターンを16進数字で表わした場合。

◎ 16進数字により、それぞれ次のドットパターンが指定されます。

16進数字	ドットパターン	16進数字	ドットパターン	16進数字	ドットパターン	16進数字	ドットパターン
0		1		2		3	
4		5		6		7	
8		9		A		B	
C		D		E		F	

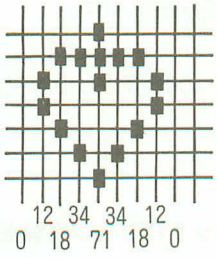
上側（4ドット）でのみこのドットパターンが指定できます。

○10進数によるドットパターン指定

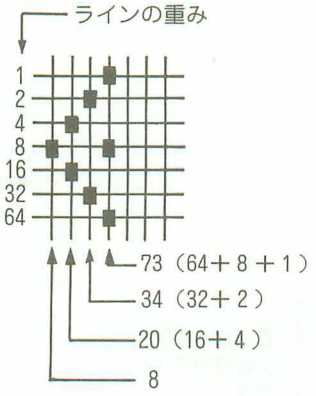
ドットパターンを10進数で指定する場合は次のようにします。

GPRINT ○;○;○;……

（例） GPRINT 12;18;34;71;34;18;12



この数値は、ドットのヨコの列（ライン）に次のような重みをつけ、表示されるドットの重みを加えることにより得ることができます。

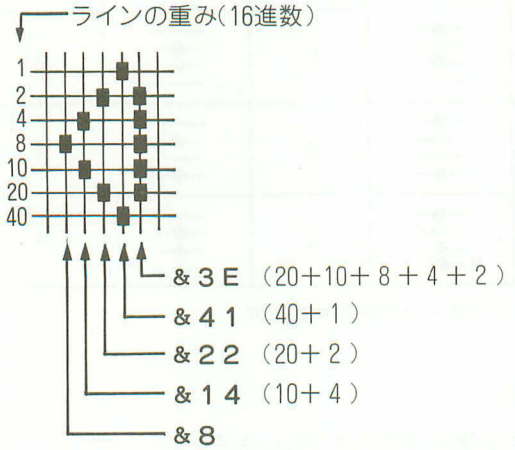


○16進数値によるドットパターン指定

10進数で指定する場合と同様の形で指定しますが、各数値の前に16進数を示す&マークをつけます。

（例） ＧＰＲＩＮＴ　＆Ｃ；＆１２；＆２２；＆４７；＆２２；＆１２；＆Ｃ

なお、ドットパターンの16進数値を得る場合は各ラインに次のように16進数値の重みをつけます。



◎ ＰＯＩＮＴ命令（ポイント）

ＰＯＩＮＴ命令は表示されている内容のドットパターンでの情報を読む命令で、指定したポジションに表示されているドットパターンを10進数値で読み出します。

ポジションはＧＣＵＲＳＯＲ命令と同様、表示部のドットの列(７ドット×156列)に左側から順に0～155までの番号をつけ、次の形で指定します。

ＰＯＩＮＴ 式

（例） Ａ＝ＰＯＩＮＴ　５　　**５番のポジションのドットパターン情報を変数Ａに入れます。**

Ｂ＝ＰＯＩＮＴ　１２０　　**120番のポジションのドットパターン情報を変数Ｂに入れます。**

（例） ◆を３個表示させ、表示全体を白黒反転させるプログラム。

```
10:GCURSOR 66
20:WAIT 0
30:FOR A=1TO 3
40:GPRINT "00081C
   3E1C0800";
50:NEXT A
60:WAIT 60:PRINT
70:WAIT 0
80:FOR J=1TO 4
90:FOR K=0TO 155
100:GCURSOR K
110:GPRINT (255-
   POINT K)
120:NEXT K
130:NEXT J
140:END
```

表示の白黒を反転させています。
（２５５は１２７でもよい。）

7．そのほかのＢＡＳＩＣ

7－1）ＲＥＡＤとＤＡＴＡ、ＲＥＳＴＯＲＥ（リードとデータ、リストア）

変数にデータを入れる（代入する）方法として、ＬＥＴやＩＮＰＵＴのほかにＲＥＡＤ命令とＤＡＴＡ命令の組み合わせがあります。

これはＲＥＡＤに続いて変数を書き、そのデータをＤＡＴＡに続いて書きます。この場合の一般形は次のようになります。

```
READ  変数,変数,……
  :
DATA  式,式,……（DATA 文字列, 文字列,……）
```

●ＤＡＴＡ文の中に、式と文字列を混ぜて書くこともできます。

（例）

```
10:READ A,B,C$
20:C=A*B
30:PRINT C$;C
40:DATA 4,20,"C="
50:END
```

Aに４を、Bに20を、
C$に"C＝"を代入します。

○上例のようにＲＥＡＤの後の変数と、ＤＡＴＡの後のデータは１対１で対応します。したがって変数の形とデータの形が一致していなければなりません。すなわち、数値変数に対応するデータは数値でなければなりませんし、文字変数に対応するデータは文字列でなければなりません。

○ プログラムの中に何回でもＲＥＡＤ文およびＤＡＴＡ文を書くことができますが、データは何ラインにわけて書いても一連のデータと見なされ、若いラインのデータから順番に変数に代入されます。

（例）

```
10:READ A,B
20:G=√(A*A+B*B)
30:READ C,D,E$,F$
40:H=C*COS D
50:PRINT E$;G;F$;
   H
60:DATA 3,4,5
70:DATA 60,"G=","
   ␣␣H="
80:END
```

Aには３、Bには４が代入されます。

Cには５、Dには60、
E$には"G＝"、F$
には"　H＝"、が代
入されます。

また、２種類以上のプログラムが書き込まれ、それぞれにＤＡＴＡ文がある場合でも、これらは一連のデータと見なされ、若いラインのデータから順番に変数に代入されます。

したがって、それぞれのプログラム内でデータを使用する場合は次に説明しますＲＥＳＴＯＲＥ命令により、代入するデータが書かれているラインを指定してください。

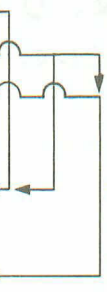
◎　ＲＥＳＴＯＲＥ命令　（リストア）

READ命令の実行時、DATA命令で指定されているデータのどのデータを読み込むかは常に計算機に記憶されていますが、この読み込むデータの順番を強制的に変えるときに使用します。

○ RESTORE 式

この形の場合、この命令を実行した次のＲＥＡＤ命令の変数には、ＲＥＳＴＯＲＥ命令の式の値で指定されるラインに書かれているＤＡＴＡ命令の最初のデータから１対１に対応して代入されます。

（例）
```
10 DATA 10, 20
20 DATA 30, 40
30 DATA 50, 60
   ⋮
100 READ A, B, C, D
110 RESTORE 20
120 READ E, F, G, H
```



A，B，C，Dにそれぞれ、10，20，30，40が代入されます。

E，F，G，Hにはそれぞれ30，40，50，60が代入されます。

○ RESTORE

この形の場合、この命令を実行した次のＲＥＡＤ命令の変数には、プログラムの最も若いラインに書かれているＤＡＴＡ命令のデータが１対１に対応して代入されます。

注１）一度実行したプログラムを再実行するとき、ＲＵＮ命令で実行すれば、ＲＥＡＤ文に対するデータの初期化がおこなわれますので、若いラインのデータから変数に代入されます。
ＧＯＴＯ命令や DEF キーに続く定義付けキーで再実行した場合はデータの初期化をおこないませんので、前回の続きのデータが代入されます。
ただし、 SHIFT CL(CA) と押したときや、 OFF ON と押して電源のオフ・オンをおこなったときはデータの初期化がおこなわれます。

注２）ＭＥＲＧＥ命令(98ページ参照)によりテープから読み込まれたプログラムをＲＵＮ〝ラベル〟により実行した場合は、そのプログラム内の若いラインに書かれているデータから代入しますが、ＧＯＴＯ〝ラベル〟や DEF キーに続く定義付けキーで実行した場合は、前回の続き、あるいは計算機内の最初のプログラムのデータを代入します。
したがって、それでは不都合な場合は、プログラムのスタートラインのラベルに続いて、「ＲＥＳＴＯＲＥ　式」の形でRESTORE命令を書いてください。

（MERGE命令により計算機内に複数のプログラムが入っているとき、READ命令により複数のプログラムからデータを読み込むことはできません。
したがって、計算機内のあるプログラムのデータを読み込もうとしたとき、そのプログラムにデータがないときや、データがたりないときはエラー（ＥＲＲＯＲ　４）になります。）

注３）ＭＥＲＧＥ命令により複数のプログラムが計算機に入っているときは、２番目以降のプログラム内のＲＥＳＴＯＲＥ命令は必ず

ＲＥＳＴＯＲＥ　式（あるいは〝ラベル〟）

の形で指定してください。
また、ＲＥＳＴＯＲＥ〝ラベル〟の形で、ほかのプログラムにあるラベルを指定したときは、そのRESTORE命令に続いてＧＯＴＯ命令を書いて、そのラインより大きいラインへ実行を移してください。

```
(例) 10 "A":DATA……                          ┐
       ⋮                                       │ プログラムA
                                               ┘
     10 "B":RESTORE "A":GOTO 20               ┐
     20 READ X                                 │ プログラムB
     30 IF X=0 THEN 20                         │
       ⋮                                       ┘
```

7−2) ON GOTO、ON GOSUB

◎ ON GOTO命令(オン・ゴートゥ)

GOTO命令についてはすでに述べましたが、ON GOTO命令はGOTO命令の機能を拡大したものです。ON GOTO命令の一般形は次のようになります。

ON 式 GOTO 式$_1$,式$_2$,式$_3$,……(ON式 GOTO "ラベル","ラベル",…)

この命令ではONの次の式の値が1の場合はGOTOに続いて書かれている式$_1$で示されるラインへジャンプし、ONの次の式の値が2の場合は式$_2$で示されるラインへとジャンプします。もし、式の値が、1より小さいときやGOTOに続いて書かれている式の数より大きい値のときは、ON GOTO命令の書かれている次のラインへ実行が移ります。

(例)

```
 10:INPUT A
 20:ON AGOTO 110,1      ←――― Aの値が1のときは110ラインへ、
   20,130                     2のときは120ラインへ、3のとき
 30:B=4,C=64                  は130ラインへジャンプします。
 40:CLS :GCURSOR C            Aの値が1より小さく、3より大
 50:WAIT 0                    きいときは次のラインへ実行が移
 60:FOR A=1TO B               ります。
 70:GPRINT "00081C
   3E1C0800";
 80:NEXT A
 90:WAIT :PRINT
100:END
110:B=1,C=74:GOTO
   40
120:B=2,C=71:GOTO
   40
130:B=3,C=67:GOTO
   40
```

注) GOTOに続く式には、二次元配列変数(A(1,2)、B$(5,3)など)を使用することはできません。

◎ ON GOSUB命令(オン・ゴーサブ)

ON GOTO命令と同じ形で指定すれば、ONに続く式の値により、GOSUBに続いて書かれた式で示すラインへサブルーチンジャンプします。

7－3）ON　ERROR（エラー処理）
（オン・エラー）

プログラムの実行中はいろいろなエラーが発生する可能性があります。エラーが発生すると計算機はエラーコードを表示して実行が止ってしまいますので、このようなときに、エラー処理をおこなうためにこのON　ERROR命令を用います。

ON　ERROR　GOTO　式

プログラムの中でON　ERRORを宣言しておきますと、以降の実行で、もしエラーが発生してもプログラムの実行は停止せずに、GOTO命令で指示されたラインへジャンプします。(文法エラー(エラーコード1番)およびINPUT命令実行時に発生するエラー（エラーコード7番、32番など)は除きます)そこでエラー処理をおこなって復帰させてやるのが通常の用い方です。

（例）
```
10:ON ERROR GOTO
   50
20:INPUT "X=";X
30:PRINT "ʃX=";ʃX
40:GOTO 20
50:PRINT "HUSUU W
   A KEISAN DEKIM
   ASEN"
60:GOTO 20
70:END
```
←――― Xの値が負数のときはエラーになり、50ラインへジャンプします。

7－4）REM（リマーク）

この命令は、プログラムの実行には関係なく、プログラムリストなどをわかりやすくする目的でプログラムの先頭や途中に注釈を入れておくものです。（REMに続いて書かれている内容はすべて注釈と見なされます。したがって、REMの後に他のステートメントを続けることはできません。）

（例）
```
10 REM KINRI KEISAN
 ⋮
200 REM SUBROUTINE
 ⋮
```

7－5）AREAD（オートリード）

この命令は定義付キーにより、プログラムの実行を開始したとき、表示されていた内容を指定された変数に読み込む命令で、定義付プログラム（39ページ参照）で、ラベルに続いて書きます。この命令の一般形は次のようになります:

AREAD　変数

（例）
```
10 "A": AREAD A
 ⋮
100 "B": AREAD A$
```

○この命令は、プログラム実行開始直後以外はスキップ(無視)されます。

○プログラムの実行を開始したとき、表示がプロンプト表示になっていたときは、指定されている変数はクリアされます。

7−6）STOP（ストップ）

この命令はプログラムの一時停止命令です。この命令を実行しますと、「BREAK IN 200」のように、実行したラインナンバーを持ったブレークメッセージを表示して実行を停止します。

（このとき、[►]、[◄]などのキーを押しますと、STOP文実行前の表示内容を呼び出すことができます。）

（例）
```
10:INPUT A,B,C,D
20:E=A*COS B-C*
   COS D
30:F=C*SIN D-A*
   SIN B
40:STOP
50:G=√(E*E+F*F)
60:H=π*(G/2)^2
70:PRINT G
80:PRINT H
90:END
```

STOP命令により実行が一時停止している状態(ブレーク状態)のときはマニュアル操作が可能になるため、変数の状態を調べたり、変数に数値を代入したりする場合によくこの命令が使用されます。

○STOP命令により停止した実行を再開させるときはCONT [ENTER] と操作してください。

7−7）ARUN（オートラン）

この命令は電源を入れたとき、自動的にプログラムの実行を開始させる命令で、プログラムの最初にこの命令を書き込んでおけば、[OFF] キーにより切られた電源を [ON] キーにより入れたときに、自動的にプログラムがスタートします。ただし、[ON] キーを押したときにモードがRUNモードになっていることが必要です。そしてオプション機器に異常がないことが必要です。

注）[OFF] キーにより電源が切られているときにオプション機器（プリンタ・カセットインターフェイス、拡張メモリーなど）をつけたり、はずしたり、またスイッチなどを切替えたりするなど、状態を変えますと、計算機はオプションに異常があると受け取ってオートランしないことがあります。このような場合はRUN命令などでプログラムを実行してください。

（例）
```
10:ARUN :WAIT 128
20:PRINT "KINRI␣K
   EISAN"
30:INPUT "GANKIN␣
   WA␣";A,"KINRI␣
   WA␣";B,"KIKAN␣
   WA␣";C
40:D=A*B/100*C/36
   5
50:WAIT :PRINT "R
   ISOKU␣WA␣";D
60:END
```

○オートランによりプログラムの実行を開始したときは、変数や各種指定（ユージング指定、カーソル指定、ウェイト指定など）は消去および解除されませんので必要に応じてCLEAR命令での変数の消去や、それぞれの命令で指定の解除などをおこなうようにプログラムしてください。

8．特別な機能

8－1）乱数関数（RND，RANDOM）

◎ **RND命令(ランダム)**

RND命令は乱数を発生させる機能を持つ関数です。

RND　X　においてXの値により次のような乱数を得ることができます。

○Xが負数の場合：　直前に発生した乱数(あるいは乱数列)と同じ乱数を発生させるために、初期値を一定にします。

○Xが0から1未満の場合：0から1未満の乱数を発生します。

○Xが1以上の場合：　1からXの値以下の乱数を発生します。なお、このときの乱数は整数値になります。

（例1）RUNモードでのマニュアル操作の場合

```
RND  0 [ENTER]    1．902513749E－01
RND  0 [ENTER]    3．757816235E－01
RND  0 [ENTER]    6．429773406E－01
```

（例2）プログラム例

```
 10:WAIT 0
 20:FOR A=1TO 3
 30:D=RND -1
 40:FOR B=1TO 4
 50:C=RND 9
 60:PRINT C;
 70:NEXT B
 80:NEXT A
 90:WAIT :PRINT
100:END
```

←この代入文は同じパターンの乱数を発生させるために入れています。
このラインを削除すれば同じパターンの乱数は発生されなくなります。
（変数Dは、Dにかぎらず、プログラム内で他に使用していない変数であればよい。）

表示例

```
9 7 1 3 9 7 1 3 9 7 1 3
```

4個の乱数を3回表示させています。

○乱数の有効桁数は10桁です。

◎ **RANDOM命令(ランダマイズ)**

上記RND命令の使用に先立って乱数のタネを植えつけるものです。

RNDは電源オンからやり直すと、常に同じ乱数を発生します。しかし、電源オン後にRANDOM命令を実行すれば同じ乱数を発生することは、まず、なくなります。

8−2）ＴＩＭＥ（タイム）

ＴＩＭＥは月日時分秒のカレンダー時計を表わす特殊な数値変数です。

○カレンダー時計の設定方法

ＴＩＭＥ＝20810 [・] 3205 [ENTER]

2月8日10時　32分5秒

○カレンダー時計の呼び出し方法

ＴＩＭＥ [ENTER]

注）このカレンダー時計は一度設定すれば、[OFF]キーなどで電源を切っても動作している24時間制の時計で、月・日も正しくカウントします。（ただし、うるう年の判断機能は持っていません。）

○ＴＩＭＥは変数と同じ扱いができますので、式の中などでも自由に使用することができます。

8−3）ＩＮＫＥＹ＄（インキードル）

プログラムの中で、この命令が実行されたとき、いずれかのキーが押されていれば、その内容を読み込んで指定された変数に代入します。

ＩＮＫＥＹ＄の一般形は次のようになります。

文字変数＝ＩＮＫＥＹ＄

（例）

```
10:RANDOM
20:WAIT 0:CURSOR
   4
30:PRINT "MARU␣KA
   ␣BATU␣KA␣?";
40:A=RND 2
50:A$=INKEY$ :IF
   A$="A"THEN 70
60:GOTO 40
70:CLS :GCURSOR 7
   4
80:WAIT 120
90:IF A=2THEN 120
100:GPRINT "1C2241
   4141221C"
110:GOTO 20
120:GPRINT "412214
   08142241"
130:GOTO 20
```

このラインをくり返し実行し、[A]キーが押されるのを待ちます。

このプログラムは、実行開始後 [A] キーを押せば○か×を表示するプログラムです。
プログラムの実行を終了するときは [BREAK ON] [CA CL] と押してください。

注1）プログラムのはじめにＩＮＫＥＹ＄がありますと、ＲＵＮ [ENTER] などでプログラムをスタートさせたとき、 [ENTER] キーを読み取ってしまうことがありますので注意が必要です。

注2）ソフトウェアキーが、ＩＮＫＥＹ＄により読み込まれた場合は アスキー コードで17～22に相当する文字（記号）が読み込まれます。（！、〃、＃、＄、％、＆ではありません）ただし、本機の場合はキャラクタコード17～22には表示できる文字や記号はありません。したがって、先のプログラムの [A] キーの替りに、たとえば [!] キーを用いる場合は、ＣＨＲ＄関数を使って、

```
50 A$=INKEY$: IF A$=CHR$ 17 THEN 70
```

のように指定します。（ＣＨＲ＄関数については74ページを参照してください。）

注3）INKEY$命令は [SHIFT] キーや [DEF] キーが押されていれば、そのキーを読み込みます。したがって、 [SHIFT] キーに続いて押したときに働く機能や入力される文字、 [DEF] キーに、続いて押したときに働く機能などを読み込むことはできません。

8－4）音発生機能（ＢＥＥＰ，ＢＥＥＰ ＯＮ，ＢＥＥＰ ＯＦＦ）

本機には音発生機能がついていますので、プログラムの途中で音を出し、計算機の操作をより効果的に、また楽しくすることができます。

◎ ＢＥＥＰ命令（ビープ）

ＢＥＥＰ命令は音の発生回数、音の高さ、長さを指定する命令です。ＢＥＥＰ命令の一般形は次のようになります。

ＢＥＥＰ　式$_1$,式$_2$,式$_3$

この一般形の式$_1$、式$_2$、式$_3$により、それぞれ次の指定をおこないます。

式$_1$：音の発生回数を指定します。式$_1$の値は0～65535の範囲内にしてください。

式$_2$：音の高さを指定します。式$_2$の値を255～0にすることにより、約230Hz～約7 kHzの周波数の音を指定することができます。

式$_3$：音の長さを指定します。式$_3$の値は0～65279の範囲内で指定してください。なお、音の長さは式$_2$の値（周波数）により、大きく変化します。

（式$_2$の値が大きくなる（周波数が低くなる）と、式$_3$の値が同じでも音が出ている時間が長くなります。）

○ 式$_3$あるいは式$_3$と式$_2$を省略することができます。式$_3$を省略した場合は、その値が160に指定され、式$_2$を省略した場合は、音の高さ（周波数）が約4 kHzに指定されます。

◎ ＢＥＥＰ ＯＮ命令（ビープオン）

この命令は音の発生機能をオンにする命令です。この命令実行後のＢＥＥＰ命令では指定された音が出ます。また、カセットの記録・再生音も発生できるようになります。

◎ ＢＥＥＰ ＯＦＦ命令（ビープオフ）

この命令は音の発生を禁止する命令です。この命令実行後はＢＥＥＰ命令による音も、カセットの記録・再生音も発生されません。

（例）

```
10:RANDOM :BEEP
   ON
20:WAIT 0:CURSOR
   4
30:PRINT "MARU_KA
   _BATU_KA_?";
40:A=RND 2
50:A$=INKEY$ :IF
   A$="A"THEN 70
60:GOTO 40
70:CLS :GCURSOR 7
   4
80:WAIT 120
90:IF A=2THEN 130
100:BEEP 2,70,50
110:GPRINT "1C2241
   4141221C"
120:GOTO 20
130:BEEP 4,140,25
140:GPRINT "412214
   08142241"
150:GOTO 20
```

（このプログラムは61ページのプログラムに音発生命令を追加したものです。）

注）ＢＥＥＰ命令は、式の値の組み合わせによっては発生する音の音質が変化することがあります。

8－5）ＣＯＮＴ（コンティニュー）

ＣＯＮＴ命令は、プログラムの実行が一時停止しているときに、プログラムの実行を再開させるための命令です。したがって、この命令はプログラムに書き込むことはできず、ＲＵＮモードでのマニュアル操作でのみ有効となります。この命令の一般形は次のようになります。

ＣＯＮＴ　[ENTER]

○プログラムの一時停止状態（ブレーク状態）とは次の状態です。

①プログラム実行中にＳＴＯＰ命令で一時停止した状態

②プログラム実行中に [BREAK ON] キーを押して一時停止した状態

③そのほか、ＰＲＩＮＴ命令の実行などにより実行が停止している状態

8－6）ＳＴＡＴＵＳ、ＭＥＭ（ステータス、メム）

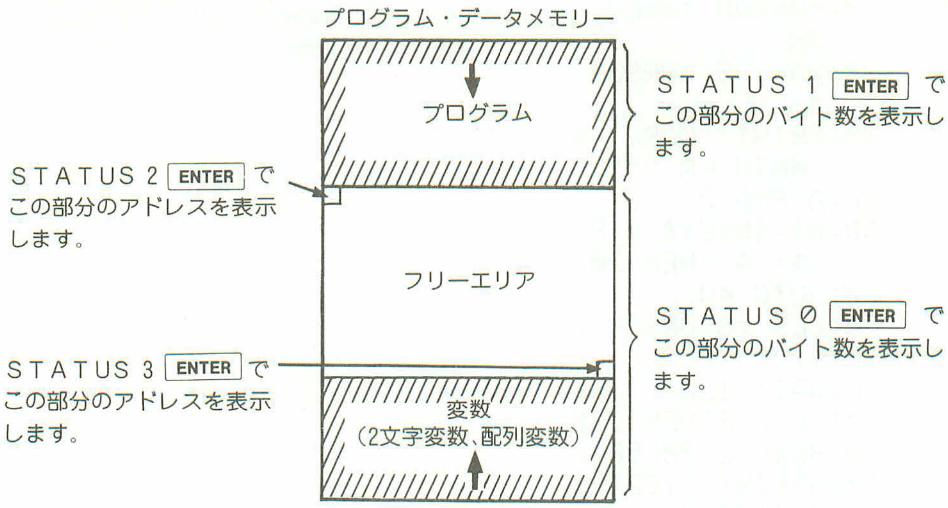
プログラムの長さや、プログラム・データメモリーの中でプログラム以外の部分のバイト数などを確認するための命令です。

◎ ＳＴＡＴＵＳ　０　[ENTER]

この命令を実行すれば、プログラム・データメモリーの プログラムが書き込まれていない部分のバイト数を表示します。（バイトとは計算機のメモリー容量を表わす単位です。）

◎ ＳＴＡＴＵＳ　１　[ENTER]

この命令を実行すれば、書き込まれているプログラムのバイト数を表示します。



◎ **STATUS 2**

この命令を実行すればフリーエリア(プログラムや変数に使用していない部分)の最初のアドレスを表示します。

◎ **STATUS 3**

フリーエリアの最終アドレスを表示します。

●フリーエリアの大きさは

(STATUS 3－STATUS 2)＋1

で求められます。

◎ **STATUS 4から255**

この命令では、直前に実行したプログラムのラインナンバーが得られます。

◎ **MEM ENTER**

この命令はSTATUS 0 ENTER と同じ働きをします。

8－7)論理積、論理和、否定(AND, OR, NOT)

◎ **論理積(AND:アンド)**

論理積は次のような値を取る関数です。

1 AND 1 ＝1
1 AND 0 ＝0
0 AND 1 ＝0
0 AND 0 ＝0

たとえば41と27の論理積(AND)を計算する場合は、これらの数値を2進数に変換して、それぞれの桁のANDを取り、その結果を10進数に変換します。

41 AND 27 ＝9

```
      101001――41
AND <
      011011――27
      ――――――
      001001――9
```

◎ **論理和（ＯＲ：オア）**

論理和は次のような値を取る関数です。

```
1  OR  1 ＝1
1  OR  0 ＝1
0  OR  1 ＝1
0  OR  0 ＝0
```

たとえば41と27の論理和（ＯＲ）はＡＮＤ命令と同様、２進数に変換して、それぞれの桁のＯＲを取りその結果を10進数に変換します。

```
41  OR  27＝59

OR＜ 101001――41
     011011――27
     111011――59
```

◎ **否定（ＮＯＴ：ノット）**

10進数を16桁の２進で表わすと次のようになります。

10進	16桁の２進
32767	0111111111111111
⋮	⋮
3	0000000000000011
2	0000000000000010
1	0000000000000001
0	0000000000000000
－1	1111111111111111
－2	1111111111111110
－3	1111111111111101
⋮	⋮
－32768	1000000000000000

ここで２進数0000000000000001の否定（ＮＯＴ）を取りますと次のようになります。

```
ＮＯＴ ＜0000000000000001
（否定）＞1111111111111110
```

このように、各桁の１を０に、０を１に反転することを“否定（ＮＯＴ）を取る”といいます。

この数値（1111111111111110）を10進数で表わす場合

```
          （２進数）      （10進数）
    1111111111111110＝？
＋）0000000000000010＝ 2
―――――――――――――――――
                   0＝ 0
```

の関係から？は－２であることがわかります。さらに

ＮＯＴ　０＝－１

ＮＯＴ－１＝０

ＮＯＴ－２＝１

となり、一般的に数値Ｘの否定は　ＮＯＴ　Ｘ＝－（Ｘ＋１）の関係式で表わせます。

8－8）条件式の結合（ＡＮＤ，ＯＲ）

これまで、ＩＦ文の中で使用しました条件式は

```
A＞5
A$＝"A"
A＊B＞＝C＋20
```

などのような、２つの値の大小比較の式が１つのみでしたが、これらの大小比較の式を２つ以上結合して、１つの条件式をつくることができます。たとえば“Ａは５よりも大きくかつ10よりも小さい”という条件を式で表わす場合、これまではＩＦ文を２つ使って

```
10  ＩＦ  A＜5  ＴＨＥＮ  30
20  ＩＦ  A＜10  ＴＨＥＮ  60
30
：
```

のような形で表わす必要がありましたが、ＡＮＤ命令を使えば

```
10  ＩＦ  A＞5  ＡＮＤ  A＜10  ＴＨＥＮ  60
```

のように１つの条件式にすることができます。

◎　ＡＮＤ命令（アンド）

２つ以上の条件のすべてを満足するような条件を１つの式で表わすときに用います。

（例）Ａは０よりも大きくかつ６より小さい場合の条件式

```
A＞0  ＡＮＤ  A＜6
```

◎　ＯＲ命令（オア）

２つ以上の条件のいずれかを満足するような条件を１つの式で表わすときに用います。

（例）　Ａは０よりも小さいか、あるいは12より大きい場合の条件式

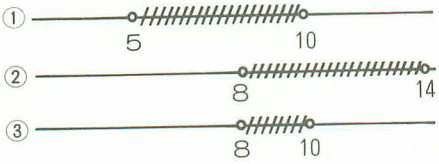
```
A＜0  ＯＲ  A＞12
```

○ＡＮＤ、ＯＲ、を使った条件式の例

①　Ａ＞５　かつ　Ａ＜10

②　Ａ＞８　かつ　Ａ＜14

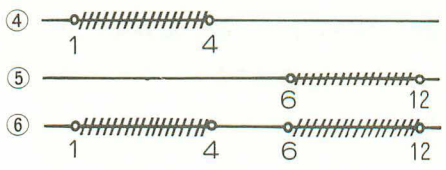
③　①と②のＡＮＤ



（Ａ＞５　ＡＮＤ　Ａ＜10）ＡＮＤ（Ａ＞８　ＡＮＤ　Ａ＜14）

この場合カッコはなくても同じです。

④ A＞1 かつ A＜4
⑤ A＞6 かつ A＜12
⑥ ④と⑤のOR



(A＞1 AND A＜4)OR(A＞6 AND A＜12)

＝参 考＝

論理計算は次のように構成することにより、論理和（OR）、論理積（AND）を形づくることができます。

① 論理和 （論理計算）＋（論理計算）

例 （A＜0）＋（A＞8） Aが0より小さいか、あるいは8より大きいときに1の値を取る

（B＞0）＋（C＞0） BあるいはCが0より大きいとき、およびB、Cともに0より大きいとき1（あるいは2）の値を取る

② 論理積 （論理計算）＊（論理計算）

例 （B＞1）＊（B＜6） Bが1より大きく、かつ6より小さいとき1の値を取る

③ 特殊な使い方の例

例 B＝（A＝1）＊3＋（A＝2）＊7 Aが1のときBは3になりAが2のときBは7になるその他のときはBは0になる

8－9）モードロック（LOCK，UNLOCK）

◎ **LOCK命令(ロック)**

MODE キーをロックして、モード(RUN、PRO、RESERVE)を他に移動させなくする命令です。たとえばRUNモードにロックすれば、MODE キーを押してもPROモードに移動することがありませんので、誤って大事なプログラムをこわしたり、消したりすることがありません。この命令の一般形は次のようになります。

LOCK ENTER

◎ **UNLOCK命令(アンロック)**

LOCK命令を解除する命令で一般形は次のようになります。

UNLOCK ENTER

8－10）OPN

別売のインターフェイスCE-158を使用するときに用いる命令で、周辺デバイスの指定をおこないます。

くわしくはCE-158の取扱説明書を参照ください。

9. デバッグ(TR ON、TR OFF)

プログラムを実行した場合に、何かの誤りで迷走したり、思わぬ結果が出てきたりすることがありますが、ここではこのようなとき、1ラインずつ経過をたどりながらプログラムを実行する方法について説明します。(このような方法をデバッグといいます。)

◎ **デバッグのしかた**

(1) デバッグはRUNモードでおこないますので、RUNモードにしてください。

(2) TR ON [ENTER] と押し、トレースモードを指定します。

(3) RUN [ENTER] と押してプログラムの実行を開始します。(最初のラインの実行が終われば実行は停止します。)

(4) その後は [↓] キーを押します。(1ラインだけ実行して停止します。)ただし、INPUT命令によるデータの入力や、PRINT命令により停止しているときの再開は、通常のプログラムの実行と同じように [ENTER] キーでおこないます。

(5) デバッグが終了すれば TR OFF [ENTER] と押してトレースモードを解除してください。

次にごく簡単な例を示します。

〈プログラム例〉

```
10 INPUT "A=";A,"B=";B
20 C=A*2
30 D=B*3
40 PRINT "C=";C;" D=";D
50 END
```

〈実行〉

```
RUNモード
TR ON [ENTER]                >
RUN   [ENTER]                A=_          ┐
      8 [ENTER] (データ入力)  B=_          ┘ INPUT命令実行
      9 [ENTER] (データ入力)  10:          ←10ライン終了
        [↓]                  20:          ←20ライン終了
        [↓]                  30:          ←30ライン終了
        [↓]                  C=16  D=27   ←PRINT命令実行
        [ENTER]              40:          ←40ライン終了
        [↓]                  >            ←実行終了
```

○デバッグ動作でラインナンバーを表示しているときは、変数の内容をマニュアル操作によって呼び出し、所望の値になっているかどうかチェックできます。また、このとき ↑ を押せば、押している間止まっているラインの内容を表示します。（なお、↑ キーを押して離したときは、ＩＮＰＵＴ命令や、ＰＲＩＮＴ命令などで表示されていた内容が表示されます。）

注）トレースモードは　ＴＲ　ＯＦＦ ENTER と押すか、SHIFT CA CL と押すまで指定された状態が保たれます。

◎ **プログラムの途中で実行を停止させてチェックする場合**

○プログラムの途中にＳＴＯＰ命令を書いておけば、プログラム実行途中でＳＴＯＰ命令を実行し、ブレークメッセージを表示して実行が停止します。
このとき

①マニュアル操作で変数の内容をチェックする。
②トレースモードを指定して ↓ キーの操作にて以降のラインを１ラインずつデバッグ操作する。
③トレースモードを解除して　CONT ENTER と操作し、通常の実行状態にて続行する。

○通常のプログラム実行中に BREAK ON キーを押しますと、現在実行しているラインの終わりで実行を停止し、ブレークメッセージを表示します。このときも前記同様３つの操作をおこなうのが普通です。なお、ブレーク状態（停止状態）のときに ↑ を押せば、押している間停止しているプログラムラインが表示されます。
（なお、↑ を離したときはＩＮＰＵＴ命令や、ＰＲＩＮＴ命令などで表示されていた内容が表示されます。）

○トレースモードを指定しない場合でも、BREAK ON キーなどにより一時停止しているプログラムを ↓ キーにより１ラインずつ実行させていくことができます。ただし、このときはラインナンバーは表示されず、ＩＮＰＵＴやＰＲＩＮＴなどの命令により表示されていた内容がそのまま表示されます。

ご注意

プログラムに書き込まれているＰＲＩＮＴ命令をＧＰＲＩＮＴ命令に訂正するときは次のことに留意してください。

（例）

```
20 PRINT A$
```

「ＰＲＩＮＴ」全体を「ＧＰＲＩＮＴ」に書き替えてください。

Ｇを挿入しただけでは、計算機には「ＧＰＲＩＮＴ」と判断できません。
（「Ｇ」と「ＰＲＩＮＴ」と見なします。）

これはＣＵＲＳＯＲ命令をＧＣＵＲＳＯＲ命令に訂正する場合や、プリンタ関係の命令についても同様です。

10. 文字の編集

この項では、文字を分断したり、結合したり、また、文字列の大小比較など、文字と文字の扱いについて説明します。

10－1）文字列の分断（ＬＥＦＴ$，ＲＩＧＨＴ$，ＭＩＤ$）

◎ ＬＥＦＴ$関数(レフトドル)

この関数は、ある文字列の左側から何文字かを取り出す関数です。この関数の一般形は次のようになります。

ＬＥＦＴ$(文字変数,数式)

あるいは

ＬＥＦＴ$(“文字”，数式)

たとえばＡ$＝"ＡＢＣＤＥ"のときＬＥＦＴ$(Ａ$,3)はＡ$の文字列の左側３文字、すなわち“ＡＢＣを取り出しなさい”という意味になります。

(例)
```
10:A$="POCKET_COM
   PUTER"
20:WAIT 30
30:FOR H=1TO 15
40:B$=LEFT$ (A$,H
   )
50:CURSOR 0
60:PRINT B$;
70:NEXT H
80:WAIT :PRINT
90:END
```

このプログラムを実行すれば表示は

```
P
PO
POC
⋮
POCKET  COMPUTER
```

のように左の文字から順番に表われてきます。

◎ ＲＩＧＨＴ$(ライトドル)

この関数はＬＥＦＴ$関数とは逆に、右側から何文字かを取り出す関数です。一般形は次のようになります。

ＲＩＧＨＴ$(文字変数,数式)

あるいは

ＲＩＧＨＴ$(“文字”，数式)

（例） 先の例のプログラムの40ラインを次のように書きかえて実行してみてください。

```
40 B$=RIGHT$(A$, H)
```

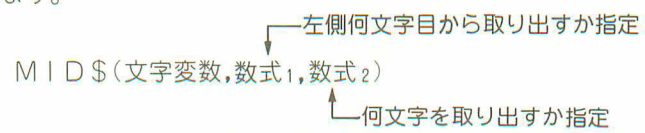
この場合は

```
R
ER
TER
 ⋮
POCKET  COMPUTER
```

のように右の文字から表われ、順順に右に送られていきます。

◎ MID$（ミッドドル）

この関数は文字列の中間の文字だけを取り出す関数です。この関数の一般形は次のようになります。



MID$(“文字”, 数式1, 数式2)

たとえば A$="ABCDE" のとき、MID$(A$,2,3)と指定した場合は、A$の左側2文字目から3文字、すなわち“BCDを取り出しなさい”という意味になります。

（例）

```
10:A$="POCKET_COM
   PUTER"
20:B=1
30:FOR H=1TO 8
40:B$=MID$ (A$,9-
   H,B)
50:B=B+2
60:CURSOR 13-H
70:WAIT 70:PRINT
   B$;
80:NEXT H
90:WAIT :PRINT
100:END
```

このプログラムを実行すると

```
          C
          CO
       T  COM
          ⋮
 POCKET  COMPUTER
```

のように真中の文字から表われます。

10－2）文字列の結合

いくつかの文字列を結合させて、新しい文字列を作る場合は、“文字列の加算”をおこないます。

たとえば、A$＝"ABC"、B$＝"DEF"、"XYZ"の3つの文字列を結合させる場合は

A$＋B$＋"XYZ"

のように加算の形で指定します。

（例）
```
10:READ A$,B$,C$,
   D$
20:E$=A$+C$+B$
30:F$=D$+B$
40:WAIT 60:PRINT
   E$;
50:WAIT 100:PRINT
   "␣␣"+F$
60:DATA "AREWA␣",
   "␣?"
70:DATA "INU","NE
   KO"
80:END
```

注）通常文字変数は16文字までを記憶することができます。文字列を変数に代入する場合は16文字を超えないようにしてください。もし、16文字を超えるような場合は文字配列変数により文字数を指定（ＤＩＭ指定）してお使いください。

10－3）文字数を数える方法（ＬＥＮ）

1つの文字列の中に含まれる文字の数（記号、スペース、数字も含みます）は計算機により、たちどころに求めることができます。

◎ **ＬＥＮ関数（レングス）**

たとえば

A＝ＬＥＮ "ＰＣ－1501"

のように指定して実行すれば "ＰＣ－1501" の文字数 7が変数Aに代入されます。この関数の一般形は次のようになります。

ＬＥＮ { “文字” / 文字変数 }

（例）
```
10:INPUT "ANATA_N
   O_NAMAE_WA_?",
   A$
20:N=LEN A$
30:FOR H=1TO N
40:B$=LEFT$ (A$,H
   )
50:CLS :CURSOR 0
60:WAIT 12:PRINT
   B$
70:NEXT H
80:GOTO 30
```

●名前を入れれば、名前の頭文字から順番に1字ずつ表示させるプログラムです。

（プログラムの実行を終わるときは BREAK ON CL と押します。）

10－4）数値と文字の変換（ＶＡＬ，ＳＴＲ＄）

数値を文字列として扱いたい場合や、逆に文字列として指定されている数字を数値として扱いたい場合がありますが、このようなとき、ＳＴＲ＄あるいはＶＡＬ関数を使います。

◎ ＳＴＲ＄関数(ストリングドル)

この関数は数値を文字列の形に変換する関数です。たとえば、Ｂ＝1234のとき、

Ａ＄＝ＳＴＲ＄　Ｂ

として実行すれば、Ａ＄には"1234"という文字列が代入されます。この関数の一般形は次のようになります。

ＳＴＲ＄　数式

（例）　Ａ＄＝ＳＴＲ＄１２０　　Ａ＄＝"１２０"と同じになります。

Ｂ＄＝ＳＴＲ＄(１．２＊３)　　Ｂ＄＝"3.6"と同じになります。

◎ ＶＡＬ関数(バリュー)

ＶＡＬ関数はＳＴＲ＄関数とは逆に文字列を数値に変換する関数です。たとえば

Ｂ＄＝"1234"のとき

Ａ＝ＶＡＬ　Ｂ＄

として実行すればＡには1234という数値が代入されます。この関数の一般形は次のようになります。

ＶＡＬ { “文字” / 文字変数 }

（例）　Ａ＝ＶＡＬ "１２０"　　Ａに120が代入されます。

Ｂ＝ＶＡＬ "３．２＊４＝"　　Ｂに3.2が代入されます。

VAL関数により数値に変換できる文字列は数値を表わすことのできる数字(0～9.)、符号(＋、－)指数部を示す記号(Ｅ)で構成されている文字列で、他の文字や記号を含んでいないことが必要です。もし、１つの文字列の中に他の文字や記号が含まれている場合は、それから右の文字列は無視されます。ただしスペースが入っていても、それはないものと見なされます。

10－5）アスキーコードの変換（ＡＳＣ，ＣＨＲ＄）

◎ アスキー(ＡＳＣＩＩ)コード

アルファベット、数字、記号などを計算機が記憶したり、処理したりする場合は、すべて計算機が取り扱いやすい数値に変換します。たとえばアルファベットのＡは計算機内では65（10進数)という数値(コード)になっています。(実際には２進数の01000001となっています。)同様にＢは66、Ｃは67というようにコードを決めています。　このコードの決めかたに何種類かあって、その１つがアスキーコード(ＡＳＣＩＩ　ｃｏｄｅ)です。

◎ ＡＳＣ関数（アスキー）

この関数は文字や記号、数字などをアスキーコードに変換する関数です。たとえば「Ｚ」という文字のアスキーコードを知りたい場合は

```
A＝ASC"Z"
```

として実行すれば、Ａには「Ｚ」のアスキーコードが10進数の90として代入されます。この関数の一般形は次のようになります。

ＡＳＣ｛“文字” / 文字変数｝

なお、文字列が２文字以上指定された場合は、頭の文字のみがアスキーコードに変換されます。

◎ ＣＨＲ＄関数（キャラクタドル）

この関数はＡＳＣ命令とは逆の関数で、10進数のアスキーコードを文字や記号（０～９の数字を含みます。）に変換する関数です。たとえば、アスキーコード「90」の文字を知りたい場合は、

```
A$＝CHR$  90
```

として実行すれば、Ａ＄には“Ｚ”が代入されます。ＣＨＲ＄の一般形は次のようになります。

```
CHR$  数式
```

なお、本機でのアスキーコードは０～255までありますが、文字や記号（これらを総称してキャラクタ：Ｃｈａｒａｃｔｅｒといいます）に対応しているコードは32～127（160～255）までです。（170ページ参照）

```
10:FOR H=32TO 127
20:WAIT 50:CURSOR
   8
30:PRINT H;"␣␣␣";
   CHR$ H
40:NEXT H
50:END
```

●このプログラムはアスキーコード32～127に対応するキャラクタを表示させるプログラムです。（ただしコード32はスペースです。）

10－６）文字列の比較

２つの文字列を比較して、その大小判断をおこなうことができます。**（アスキーコードの大きさによる大小判断）**この判断を用いれば、たとえば文字列をアルファベット順に並べ替えることも容易になります。　文字列の比較は、数値の比較に用いました等号、不等号、大小比較の記号を用いることによりおこないます。たとえば

```
"ABC"＜"BC"
"AKIRA"＞"AKIO"
"YAMATO"＝"YAMATO"
```

（アスキーコードではＡが65、Ｂが66、Ｃが67……となっています。したがって、文字列の比較では、ＡはＢより小さく、ＢはＣよりも小さくなります。また、数字はアルファベットよりも小さくなります。）

のように書き表わすことができます。いま、文字列をＡ＄とＢ＄とした場合、次の種類の比較ができます。

```
A$＝B$    A$とB$は同じ文字列です。
A$＜B$    A$はB$よりも小さい文字列です。
```

A$>B$　A$はB$よりも大きい文字列です。
A$<=B$　A$はB$よりも小さいか等しい文字列です。
A$>=B$　A$はB$よりも大きいか等しい文字列です。
A$<>B$　A$とB$は同じ文字列ではありません。

これらはＩＦ文の中に使用して大小判断に用いることができます。

（例）　５人の名前を入れて、アルファベット順に並べ替えるプログラム。

```
10:DIM A$(4)*10
20:FOR H=1TO 5
30:WAIT 0:CLS
40:PRINT "NAMAE";
   H;"___";
50:INPUT A$(H-1)
60:NEXT H
70:FOR B=0TO 3
80:FOR C=B+1TO 4
90:IF A$(B)<=A$(C
   )THEN 110
100:D$=A$(B):A$(B)
   =A$(C):A$(C)=D
   $
110:NEXT C
120:NEXT B
130:"A":FOR J=1TO
   5
140:WAIT 140:CLS
150:PRINT "NO.";J;
   "___";A$(J-1)
160:NEXT J
170:END
```

入力ループ

比較・並べ替えループ

並べ替え

出力ループ
（ DEF A と押せばこのループのみ実行）

上記プログラムを実行してください。データは次の通りです。

ＨＩＲＯＳＩ	ＡＫＩＫＯ
ＳＡＴＵＫＩ	ＡＫＩＯ
ＡＫＩＫＯ　→	ＨＩＲＯＳＩ
ＫＡＺＵＭＩ	ＫＡＺＵＭＩ
ＡＫＩＯ	ＳＡＴＵＫＩ

プログラムを実行し、左の名前を入れれば右のように並べ替えられます。

＝式の中での文字列の長さについて＝

文字列の結合（加算）や、大小比較などの式において、文字列の長さ（文字数）の合計は80文字以内でなければなりません。80文字を超えますと文字記憶エリア（計算や比較のため、文字を一時記憶しておくところ）の容量をオーバーしてエラーになります。（ＥＲＲＯＲ　15）

たとえば、次の式では文字列の長さの合計は９文字になります。

```
X=LEN("ABC"+LEFT$("DEFGHI",2))
```

11. リザーブ機能

[Q] [W] …… [P] キーには、それぞれにＢＡＳＩＣの命令が割り当てられていて、各命令を簡単に入力することができるようになっていますが、ほかにＢＡＳＩＣの命令や、よく使う計算式などを自由に割り当てることのできるキーがあります。
キーボードには、キー上になにも書かれていないキーが６個ありますが、このキー（ソフトウェアキー）に必要な命令や計算式などを割り当て、記憶させておく(リザーブする)ことができ、その内容を利用することができます。

たとえば [!] ["] [#] …キーにそれぞれＳＩＮ、ＣＯＳ、ＴＡＮ、……などをリザーブしておけば、ＳＩＮ30＋ＣＯＳ30を入力するときは

[!](SIN) 30 [+] ["](COS) 30

と操作するだけで済みます。
なお、各キーにリザーブした内容を示すため、リザーブした内容に合わせて、それらを示す文字や記号など(キーシンボル)を計算機に記憶させ、必要に応じて表示部に表示させることもできます。

ソフトキーは６キー有り、１キーごとにリザーブしますが、リザーブしたい内容が７種類以上あるときは６キーでは足りません。このような場合には、表示部の右上に表示されているリザーブナンバーを [▲▼] で切替えれば、それぞれのキーに重ねてリザーブすることができます。
リザーブナンバーはⅠ、Ⅱ、Ⅲと３つあります。したがって１つのキーには３種類の内容までリザーブすることができます。

11－１）リザーブのしかた

◎ **準備**

① ＲＥＳＥＲＶＥモードを指定します。 [SHIFT] [MODE] と押して表示部にＲＥＳＥＲＶＥシンボルを表示させてください。

② すべての内容を新しくリザーブするときは、ＮＥＷ [ENTER] と押し、前にリザーブされていた内容をすべて消してください。ただし、前の内容を残しておきたいときはこの操作は不要です。

●リザーブ内容をすべて消したときは、 [SHIFT] キーを押さなくてもソフトウェアキーのみで ！、〃、＃…などの記号を入力することができます。

◎ **リザーブ**

次の手順でリザーブします。

① [▲▼] キーを押して、表示部右上にⅠ、Ⅱ、Ⅲのいずれかのリザーブナンバーを表示させてください。(たとえばⅠを表示させてください。)

② そののち、次の形でリザーブしてください。

[ ] リザーブしたい内容を入力 [ENTER]

└ソフトキー([!] ["] [#] [$] [&] [%])

③ リザーブする内容が多くあり、1つのリザーブナンバーの中で、各キーにリザーブし切れないときは [▲▼] を押し、ナンバー表示を替えて、リザーブしてください。

（例）

[SHIFT] [MODE]　（RESERVEシンボルを表示させてください。）

（NEW [ENTER]）

[▲▼]　（リザーブナンバーⅠを表示させてください。）

[!]	SIN [ENTER]	F1：SIN
[II]	COS [ENTER]	F2：COS
[#]	A＊A＋B＊B [ENTER]	F3：A＊A＋B＊B

リザーブナンバーⅠでのリザーブ

[▲▼]　（リザーブナンバーⅡを表示させてください。）

[&]	RUN [ENTER]	F6：RUN
[%]	WAIT [ENTER]	F5：WAIT
[!]	FOR [ENTER]	F1：FOR
[II]	TO [ENTER]	F2：TO
[#]	STEP [ENTER]	F3：STEP
[$]	NEXT [ENTER]	F4：NEXT

リザーブナンバーⅡでのリザーブ

○ソフトキーには左から、それぞれF1、F2、…F6のキーナンバーがつけられています。キーナンバーは、今どのキーが指定されているか、あるいはリザーブ内容がどのキーにリザーブされているものかを示します。

○本機はこのリザーブのために、110バイトの容量が用意されています。リザーブはキーナンバーも含めて、110バイト以内でおこなってください。
次にそれぞれの文字や命令などのバイト数の例を示します。

① キーナンバー……1バイト
（例） F1：、F2：……

② 数字・符号……1バイト
（例） ＋、－、1～0……E、√、π

③ 関数・ベーシック命令……2バイト
（例） SIN、COS、SQR、PI、INPUT、RUN、IF……

○1つのキーにリザーブできる内容の長さはキーナンバーや [ENTER] キーを含めて最大80キー命令です。

● キー命令はおおむね表示1桁に対し、1キー命令となります。（ソフトキーはキーナンバーに表示3桁を使用している通り、3キー命令になります。）

（例） [!] SINA＋COS B [ENTER]　　F1：SIN A＋COS B

13キー命令　　8バイト

◎ リザーブ内容の訂正・変更

リザーブ内容の訂正や変更は、マニュアル操作で説明しましたキー操作の訂正やプログラムを修正するときと同様の方法でおこないます。

（例） リザーブナンバーⅠ、キーナンバーF2(II キー)にリザーブされているCOSをACSに変更したいとき

[▲▼] を操作してⅠシンボルを表示させます。

[II] F2：COS

[►] または [◄] を操作 F2：COS

ここを点滅させます。

ACS [ENTER] F2：ACS

◎ リザーブ内容を消去したいとき

○リザーブした内容をすべて消去したいときは

RESERVEモードで NEW [ENTER]

と押してください。

○1つのキーにリザーブされている内容を消去したいときには、次の例のようにリザーブ内容を [SPACE] あるいは [SHIFT] [DEL ◄] で消してください。

（例） リザーブナンバーⅠ、キーナンバーF3(# キー)にリザーブされているA＊A＋B＊Bを消去します。

[▲▼] を操作してⅠシンボルを表示させます。

[#] を操作 F3：A＊A＋B＊B

[►] または [◄] を操作 F3：A＊A＋B＊B

[SPACE] キーあるいは F3： _

[SHIFT] [DEL ◄] を操作して内容をすべて消します （F3：_ ）

[ENTER] F3：

◎ リザーブした内容の利用のしかた

リザーブした内容はRUNモードやPROモードで、計算式やプログラムを書き込むときなどに利用します。

（例） 先に、ソフトキーにリザーブした内容を用いてプログラムを書き込んでみましょう。

[▲▼] でⅡシンボルを表示させてください。

PROモードにしてください。

! を押す　" を押す　# を押す

```
10 FOR A=0 TO 90 STEP 5
20 B=SIN A
30 WAIT 60
40 PRINT "SIN_";A;"=_";B
50 NEXT A
60 END
```

（30 WAIT は % を押す、50 NEXT は $ を押す）

このようにキーの操作が、かなり少なくなります。

次にRUNモードにして、いま書き込んだプログラムを実行してみましょう。

ここでも [&] キーにリザーブされているRUN命令を用いれば

[&] [ENTER]

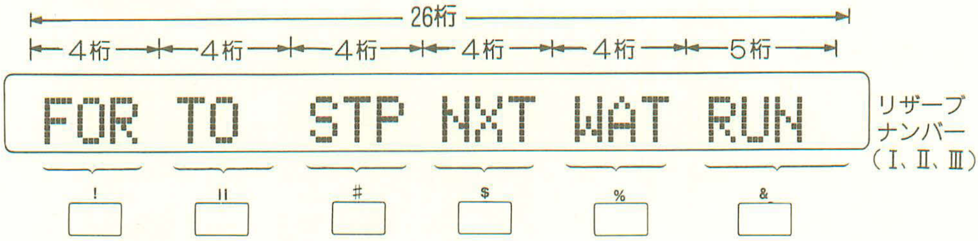
と押すだけで実行を開始します。

なお、たとえばRUN@のようにリザーブしておけば@が [ENTER] キーと同じに働きますので、ソフトウェアキーだけで実行を開始させることもできます。

11―2）ソフトウェアキーのキーシンボル表示

ソフトウェアキーにリザーブした内容がすぐにわかるようにするため、それぞれのキーの位置に、リザーブした内容を表わす文字や記号をキーシンボルとして表示させるようにします。（下図参照）

◎ リザーブキーと表示位置



このような形で文字や記号を表示させます。

◎ キーシンボルの書き込みかた

キーシンボルは、表示される内容すべてを１つの文字列として書き込みます。したがって、先の表示の例では、次のようにＦＯＲからＲＵＮまですべてを“　”で囲って書き込みます。（“　”内はスペースも含めて26字以内にしてください。）

（例）[SHIFT] [MODE] と押してＲＥＳＥＲＶＥシンボルを表示させます。

[▲▼] キーを押してリザーブナンバーⅡを表示させます。

"␣ＦＯＲ␣ＴＯ␣␣ＳＴＰ␣ＮＸＴ␣ＷＡＴ␣ＲＵＮ" [ENTER]

このとき、リザーブした内容のリザーブナンバーと、キーシンボルを書き込むときのリザーブナンバーを一致させておかなければ、キーシンボルと、リザーブされている内容が異ってしまいますので、必ずリザーブナンバーを一致させてください。

ソフトキーには３種類の内容がリザーブできますので、それに合わせて、キーシンボルも３種類書き込んでください。先に述べましたように、リザーブナンバー（Ⅰ、Ⅱ、Ⅲ）を一致させることを忘れないでください。

（例） 前の例に続いてナンバーⅠのキーシンボルを書き込みます。

[▲▼] [▲▼] と押してリザーブナンバーⅠを表示させます。

"␣ＳＩＮ␣ＡＣＳ" [ENTER]

○キーシンボルは、あなたにとってわかりやすければ、どのような文字、記号を用いてもかまいません。キーシンボルは、単にリザーブされている内容を呼び出すときの目安となるものです。

○キーシンボルを新たに書き込めば、前に書かれていたシンボルは消されます。

注）キーシンボルを書き込むとき、“　”内は26字以内（スペースを含む）にしてください。

“　”内に27字以上入れた場合は先頭の26字が有効になります。

◎ キーシンボルの利用のしかた

（例） 先にリザーブナンバーⅡにリザーブした内容とキーシンボルを利用して、次のプログラムラインを書き込みます。

```
10 FOR A=0 TO 90 STEP 5
20 B=SIN A
```

PROモード

ナンバーⅡを表示させます。(キーシンボルが表示されます。)

[▲▼]　　FOR TO STP NXT WAY RUN

10 [!] A=0　　10FOR A=0_

キーシンボルを呼び出してキーを探します。

[RCL]　　FOR TO STP NXT WAY RUN

[=] 90　　10 FOR A=0TO 90_

キーシンボルを呼び出してキーを探します。

[RCL]　　FOR TO STP NXT WAT RUN

[#] 5 [ENTER]　　10:FOR A=0TO 90STEP 5

20 B=　　20B=_

ナンバーⅠを表示させます。(キーシンボルが表示されます。)

[▲▼] [▲▼]　　SIN ACS

[!]　　20B=SIN_

A [ENTER]　　20:B=SIN A

○ [RCL] キーはキーシンボル表示と、現在表示されている内容を入れ替えます。

○キーシンボルの呼び出しは、必要な内容がリザーブされているキーを探すためですので、既にわかっている場合は必要ありません。

＝参　考＝

ソフトキーを定義付けキーと同じように使用して、プログラムの実行を開始させることができます。たとえば [!] に GOTO 100@ とリザーブしておけば、RUNモードで [!] を押すだけで100ラインへジャンプして、プログラムの実行を開始します。(@が [ENTER] キーの働きをします。)

カラーグラフィックプリンタについて

本機は、 別売のカラーグラフィックプリンタ〈CE-150〉(カセットインターフェイス内蔵)を接続し、使用することができます。 CE-150をお使いいただくことにより、計算結果やプログラムを書き出したり、図形を書いたりすることができます。また、CE-150をお使いいただくことにより、テープレコーダーを外部記憶装置として使用することができ、カセットテープなどにプログラムやデータなどを記憶したり、その内容をテープから読み込んだりすることができます。テープレコーダーは2台接続することができますので、 一方を読み込み用、 もう一方を記録用として、 ばらばらに記録されているプログラムを一本のテープにまとめたり、 一方からデータを読み込んで処理し、もう一方に記録していくなど、応用範囲も広がります。

〈CE-150と計算機の接続〉

カラーグラフィックプリンタ CE-150と本機を接続するときは、次の手順でおこなってください。

(1) OFF キーを押して計算機の電源をオフにしてください。

(2) 計算機の左側面にある端子カバーをはずし、プリンタの裏面の保存場所に取りつけてください。

図1

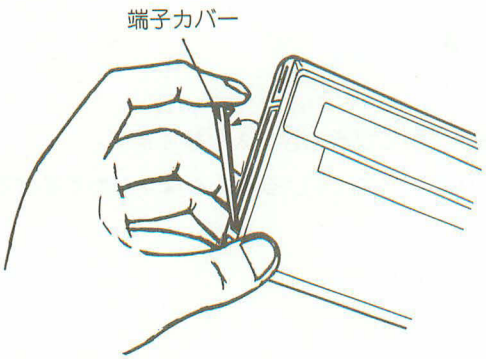
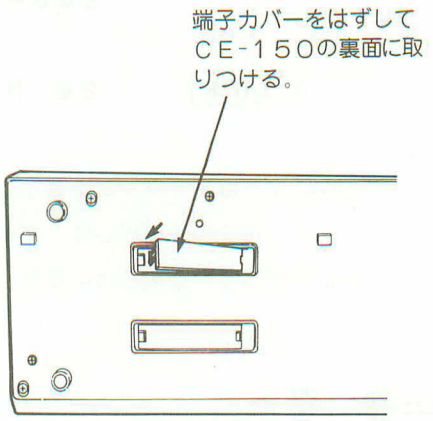
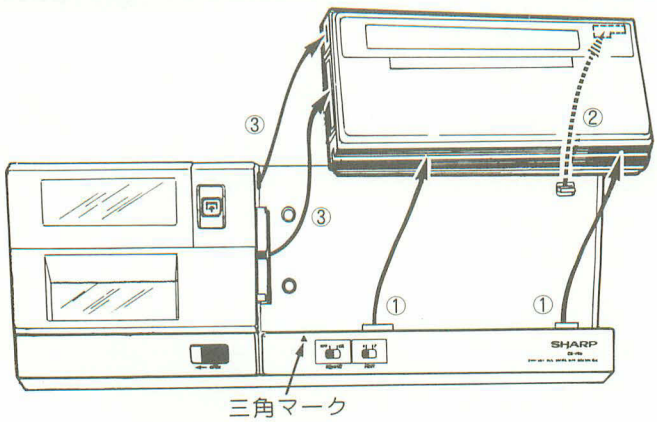


図2



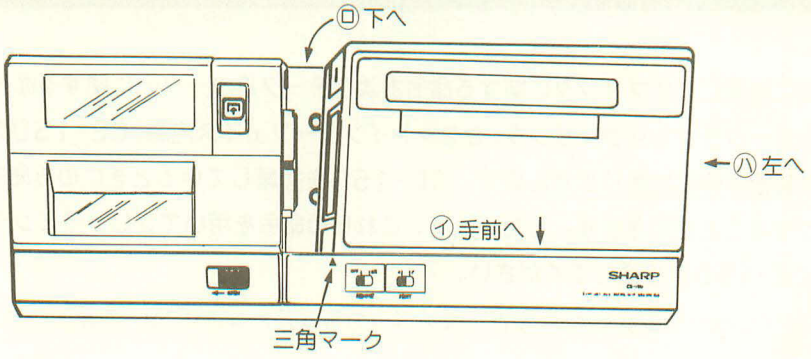
(3) 図の矢印のようにプリンタの突起部分を計算機のミゾ部分に入れていきますが、入れていく順序は矢印の番号および次の説明順序通りです。

図3



① ▲の部分に計算機の左側面がくるように計算機をプリンタに乗せてください。

図4



②図3の矢印②の部分が入らないときは、計算機を右側、あるいは左側へ軽く動かしてください。

③矢印②の部分が入ったら図3の矢印③の部分(計算機とプリンタ)を差し込んでください。

●接続端子は確実に差し込んでください。ただし、むりやり差し込まないでください。

注) 計算機をプリンタに取りつけるとき、あるいは取りはずすときは [OFF] キーにより計算機の電源をオフにしておこなってください。

電源をオンのままで取りつけたり、取りはずしたりしたときは計算機のすべてのキーが働かなくなることがあります。万一このような状態となったときは [ON] キーを押しながら計算機裏面のオールリセットスイッチを押してリセットしてください。このとき計算機の内容はすべてクリアされますのではじめから操作してください。

注) 計算機とプリンタが接続されている場合に、 [ON] キーにより計算機の電源を入れたとき、

NEW0?:CHECK 6

あるいは

:CHECK 6

のようにチェックメッセージが表示されることがあります。また、使用中にERROR 80とエラーメッセージが表示されることがあります。

これらは、プリンタの電池の電圧低下などにより、プリンタが正常に動作できないことを示しています。このような場合は、CE-150の取扱説明書をご覧のうえ、充電をおこなってください。

なお、上記メッセージが表示された場合はプリンタがロックされます。これを解除するには充電をおこなったのち、 [OFF] [ON] と押してください。

プリンタに関する命令

これ以降に説明しますプリンタに関する命令およびテープレコーダーに関する命令(96ページ)は、別売のカラーグラフィックプリンタ(カセットインターフェイス内蔵)CE-150が持っている命令で、計算機本体にはありませんので、CE-150を接続しているときにのみ命令としてプログラミングすることができます。したがって、これらの命令を用いてプログラミングするときは、必らずCE-150を接続してください。

(注) プリンタのペン交換の方法や、ペーパーのつけかたなどについてはCE-150の取扱説明書をご覧ください。

(1) モード指定命令(TEXT, GRAPH)

本プリンタは、計算結果や、プログラムリストなどの文字や数字を印字する場合と、グラフなど図形を書く場合のように、その使用目的により2つのモードがあります。

◎ TEXT命令(テキスト)

文字や数字を印字するモード(TEXTモード)を指定する命令で、一般形は次のようになります。

TEXT

(TEXT [ENTER])

なお、[OFF] キーで切られた電源を [ON] キーで入れたときはTEXTモードが指定されます。また、後で述べますLLIST命令、TEST命令を実行した場合やマニュアル操作で計算結果などを印字した場合は自動的にTEXTモードが指定されます。

◎ GRAPH命令(グラフィック)

グラフなどの図形を書くモード(GRAPH)を指定する命令で、一般形は次のようになります。

GRAPH

(GRAPH [ENTER])

このモードが指定されますと、ペンを左端にもどして、その位置をこれから図形などを書く場合の原点にします。

(2) プリント命令(LPRINT)

◎ LPRINT命令(ラインプリント)

計算機の表示命令として、PRINT命令がありましたが、これに替わるプリンタの印字命令としてLPRINT命令があります。

このLPRINT命令の一般形はPRINT命令とほとんど同じ形で指定することができます。

① LPRINT { 式 / 文字変数 / "文字" }

○TEXTモードにおいて、ペンが左端にあるときは、数値を右づめで、文字を左づめで印字します。もし、ペンが左端にないときは、文字・数値ともそのペン位置から印字されます。

② LPRINT { 式 / 文字変数 / “文字” } , { 式 / 文字変数 / “文字” }

○TEXTモードにおいて、文字サイズ1(1行36桁)のときは左右2分割して、指定された内容を印字します。ただし、数値が1行に印字可能な桁数を超えた場合はエラーになります(ERROR 76)

文字が1行に印字可能な桁数を超えた場合は、超えた部分は無視されます。

(1行に印字できる桁数については下の表を参照ください。)

③ LPRINT { 式 / 文字変数 / “文字” } ;

○TEXTモードにおいて、現在のペン位置から印字をおこないます。印字内容が1行に印字できないときは改行して印字されます。

(3) 文字指定命令(CSIZE, ROTATE, COLOR)

文字の大きさや文字の向き、また何色で印字するかを指定する方法を説明します。

◎ CSIZE命令(キャラクタサイズ)

文字の大きさを指定する命令で、次のような形で指定します。

CSIZE 式

式の値を1～9にすることにより印字する文字の大きさを指定します。

それぞれの値に対する文字の大きさ、横1行に印字できる桁数は次の表のようになります。

CSIZE	1	2	3	4	5	6	7	8	9
桁数	36	18	12	9	7	6	5	4	4
文字高さ(mm)	1.2	2.4	3.6	4.8	6.0	7.2	8.4	9.6	10.8
文字幅(mm)	0.8	1.6	2.4	3.2	4.0	4.8	5.6	6.4	7.2

●TEXT命令およびGRAPH命令が実行されますと、文字サイズは自動的に2が指定されます。

●CSIZE 1(もっとも小さい文字)が読み取りにくい場合は CSIZE 2 をご使用ください。

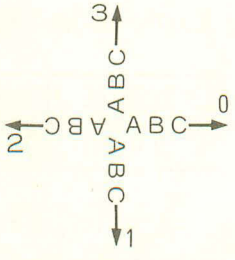
◎ ROTATE命令(ローテイト)

ROTATE命令は文字の向き(印字方向)を指定する命令で、次のような形で指定します。

(**GRAPHモードでのみ有効**な命令です。)

ROTATE 式

式の値を0～3にすることにより、文字方向が右に示すようになり、矢印方向にむかって印字がおこなわれます。



●TEXT命令およびGRAPH命令が実行されますと、文字の向きは自動的に 0 の方向が指定されます。

◎ COLOR命令(カラー)

COLOR命令は印字する文字や線の色を指定する命令で、次のような形で指定します。

COLOR 式

式の値を0～3にすることにより、文字などの色を指定します。文字の色は黒、青、緑、赤の4色あり、それぞれの色を指定します。

●プリンタには4本のプリンタ用ボールペンを取りつけます。このボールペンの色を黒、青緑、赤にすれば、COLOR命令により、指定された位置に取りつけられている色で印字をおこなうことができます。なお、電源を入れたときはCOLOR　0に相当する位置に取りつけられている色(ボールペン)が選ばれます。

(4) 色やインキのチェック(TEST)

◎ TEST(テスト)

色の指定のチェック(COLOR命令で0～3に指定される色のチェック)やインキ量、インキの乾燥をチェックするための命令で、次のような形で指定すれば、5㎜×5㎜の正方形をそれぞれの色で書きます。

TEST

(TEST ENTER)

□ □ □ □

0位置の色　1位置の色　2位置の色　3位置の色

(5) 印字位置の指定(TAB, LCURSOR)

◎ TAB命令(タブ)

計算機のCORSOR命令が、表示の桁単位で表示開始位置を指定するのと同様、TAB命令はペンの位置を指定のポイントまで移す命令で、次のような形で指定します。**(TEXTモードでのみ有効)**

① TAB 式

② LPRINT TAB 式$_1$; 式$_2$……

TABに続く(式$_1$)の値によりペンの位置を指定します。ただし、文字の大きさにより、ペンの移動量が異なりますので、式の値の範囲も次の表に示す通り、変化します。

文字の大きさ指定	CSIZE 1	CSIZE 2	CSIZE 3	CSIZE 4	CSIZE 5	CSIZE 6	CSIZE 7	CSIZE 8	CSIZE 9
式の範囲	0～35	0～17	0～11	0～8	0～7※	0～5	0～5※	0～4※	0～3

※CSIZE 5、7、8において、それぞれTAB7, TAB5, TAB4を指定した場合は続いて印字をおこなうとき、改行あるいはエラーになります。

○次のように指定した場合は１行改行される命令になります。

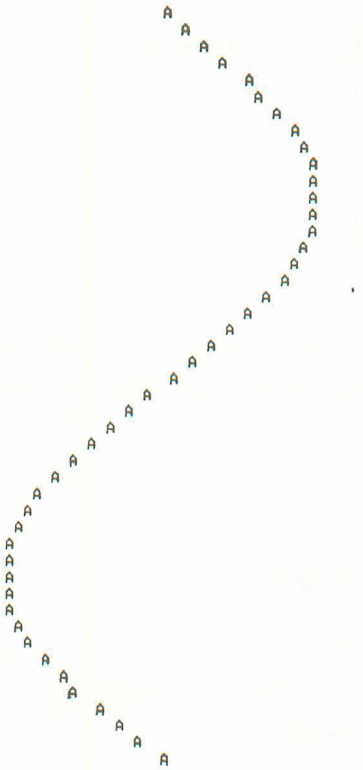
LPRINT　TAB　式;　　１行改行を指定

印字する内容を指定しません。

（例）　文字Ａを用いてサインカーブを書いてみましょう。

```
10:TEXT :CSIZE 1
20:FOR H=0TO 360
   STEP 8
30:A=SIN H
40:B=INT (A*17)
50:TAB (B+18)
60:LPRINT "A"
70:NEXT H
80:END
```

RUN [ENTER]



◎　ＬCURSOR

ＬCURSOR命令の働き、一般形はＴＡＢ命令と同じです。

(6)　紙送り命令（ＬＦ）

◎　ＬＦ命令（ラインフィード）

ＬＦ命令は紙送りをおこなう命令で、次の形で指定します。（**ＴＥＸＴモードでのみ有効**）

ＬＦ　式

式の値で示す行数の紙送りをおこないます。（式の値は32767までです。）

式の値が負のときは逆方向へ最大10.24 cmまで紙送りをおこないます。

(7) ペン移動命令(ＧＬＣＵＲＳＯＲ)

◎ ＧＬＣＵＲＳＯＲ命令(グラフィックラインカーソル)

ペンの位置を原点からＸ軸、Ｙ軸方向に移動させる命令で、次のような形で指定します。

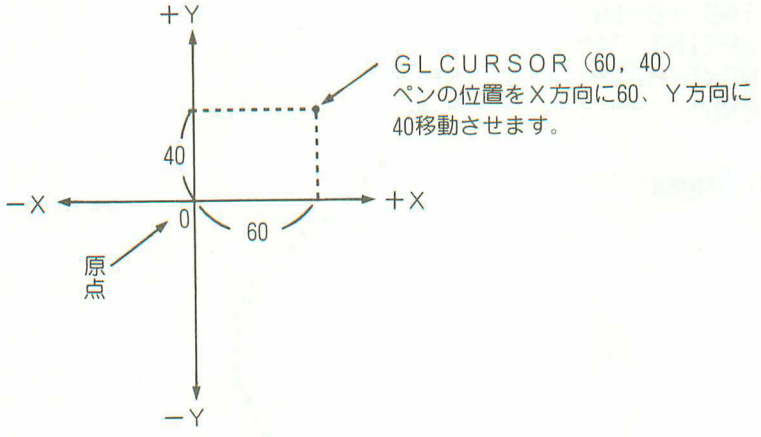
(ＧＲＡＰＨモードでのみ有効)

ＧＬＣＵＲＳＯＲ (式1, 式2)

式1の値でＸ軸方向へ、式2の値でＹ軸方向への移動量を指定します。

式の値は－2048～＋2047まで指定可能で、ペンの最小移動量は0.2㎜になります。

(例)



注) ＋Ｙ方向にペンを移動させることはペーパー(紙)を逆方向に送ることになります。この場合は最大10.24 cmまでしか送られませんので注意してください。

(8) 原点指定命令(ＳＯＲＧＮ)

◎ ＳＯＲＧＮ(セットオリジン)

現在あるペンの位置を新しい原点に指定する命令で、一般形は次のようになります。

(ＧＲＡＰＨモードでのみ有効)

ＳＯＲＧＮ

ＧＲＡＰＨモードが指定されたときはペンが左端にもどり、その位置が原点になります。図形を書く場合などでは、原点が左端にあると書きにくい場合などがありますので、たとえばＧＬＣＵＲＳＯＲ命令で、任意の位置にペンを移動させ、ＳＯＲＧＮで原点指定をして、その位置を基点として図形などを書きます。

(9) 描線命令(ＬＩＮＥ，ＲＬＩＮＥ)

指定位置から指定位置まで線を書く命令です。

◎ ＬＩＮＥ命令(ライン)

次の形で線を書く位置、線の種類、色などを指定します。**(ＧＲＡＰＨモードでのみ有効)**

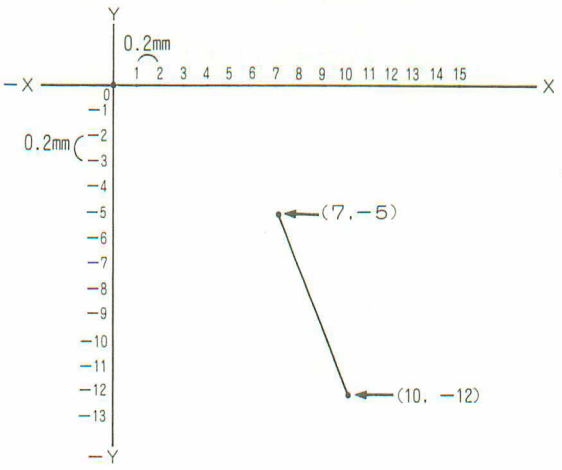
〈一般形〉

ＬＩＮＥ(式$_1$，式$_2$)ー(式$_3$，式$_4$)，式$_5$，式$_6$
　　Ⓐ　　Ⓑ　　Ⓒ　Ⓓ

○線の位置、長さ、方向の指定

(式$_1$，式$_2$)で示される座標位置から、(式$_3$，式$_4$)で示される座標位置まで線を書きます。

なお、(式$_1$，式$_2$)を省略することができます。この場合はペン位置から(式$_3$，式$_4$)で示される位置まで線を書きます。



○線の種類

式$_5$の値によりそれぞれ次のように線の種類を指定します。

0：実線

1～8：破線

9：線を書かずにペンを移動

(例)

1 ----------

2 ----------

3 ----------

4 ----------

5 ----------

6 ----------

7 ----------

8 ----------

○線の色

式$_6$の値により線の色を指定します。式$_6$の値は0～3で指定します。

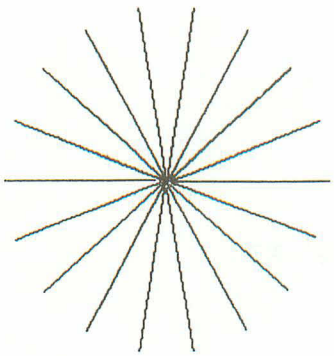
●線の色については86ページのＣＯＬＯＲ命令を参照してください。

なお、式$_5$、式$_6$は省略することができます。式$_5$や式$_6$が省略されたときは、それ以前に指定されている値(状態)で実行します。

(例)

```
10:GRAPH :RANDOM
20:GLCURSOR (107,
   -120):SORGN        ←図形を書くため、ペン位置をペーパーの中ほどに
                        移動させて、その位置を原点に指定します。
30:FOR J=0TO 340
   STEP 20
40:A=107*COS J
50:B=107*SIN J
60:R=RND 4-1           ←0～3の乱数をRに代入
70:LINE (0,0)-(A,
   B),0,R              ←Rの値により色を指定します。
80:NEXT J
90:END
```

RUN [ENTER]



○四角形を書くには

ＬＩＮＥ命令の最後にＢを入れることにより、(式$_1$，式$_2$)で示される座標位置と(式$_3$，式$_4$)で示される座標位置を対角線とする四角形を書きます。ただし、式$_1$と式$_3$あるいは式$_2$と式$_4$が同じ値の場合は四角形にならず、直線が書かれます。

(例)　ＬＩＮＥ(10,50)－(200,－20),2,0,Ｂ

○連続して線を書く場合の指定のしかた

次のようにＬＩＮＥ命令を書くことにより、連続した線をつぎつぎに書くことができます。

ＬＩＮＥ(式$_1$，式$_2$)ー(式$_3$，式$_4$)ー(式$_5$，式$_6$)ー……(式$_{11}$，式$_{12}$)，式，式

このように指定すれば(式$_1$、式$_2$)の点から(式$_3$，式$_4$)の点まで線を書き、続いて(式$_3$，式$_4$)の点から(式$_5$，式$_6$)の点まで線を書く……というように連続して線を書くことができます。なお、(式，式)は６つまで指定することができます。

◎ **ＲＬＩＮＥ命令(レラティブライン)**

この命令の一般形はＬＩＮＥ命令と同じ形になります。

〈一般形〉

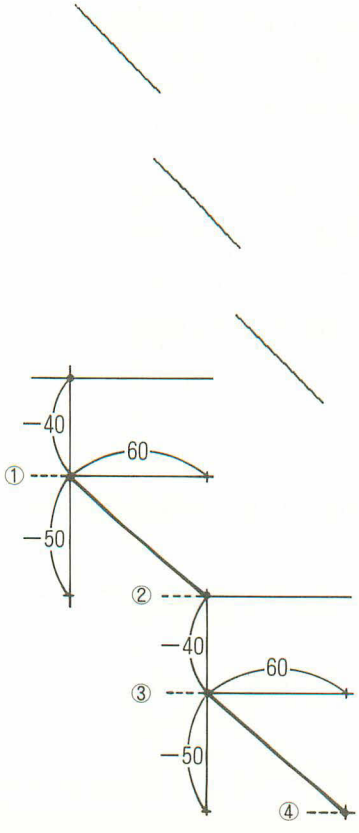
ＲＬＩＮＥ(式$_1$，式$_2$)ー(式$_3$，式$_4$)，式$_5$，式$_6$

Ⓐ　　Ⓑ

ＬＩＮＥ命令とＲＬＩＮＥ命令との違いは、ＬＩＮＥ命令が、ＳＯＲＧＮ命令などにより指定された原点を基準にし、その原点からの座標により各点の位置を指定するのに対し、ＲＬＩＮＥ命令は、ペン位置を原点にしてその位置からの座標により次の点の位置を指定します。そのほかはＬＩＮＥ命令と同様の形で使用します。

(例)
```
10:GRAPH
20:FOR A=1TO 3
30:RLINE (0,-40)-
   (60,-50)
40:NEXT A
50:END
```

このプログラムを実行すれば次のようにプリントされます。



①の位置を原点として②の位置を決め、
②の位置を原点として③の位置を決めます。
そして③の位置を原点として④の位置を決めています。

⑽ リストのしかた(ＬＬＩＳＴ)

◎ ＬＬＩＳＴ命令(ラインリスト)

この命令は計算機に記憶されているプログラムを書き出す命令で、一般形は次のようになります。(**ＰＲＯあるいはＲＵＮモードでのマニュアル操作により実行します。**)

ＬＬＩＳＴ式$_1$，式$_2$ **ENTER**

この形により、式$_1$の値で示されるラインから、式$_2$の値で示されるラインのプログラムまでが書き出されます。なお、式$_1$、式$_2$は省略することができ、これらを省略した場合はそれぞれ次のようになります。

ＬＬＩＳＴ **ENTER**

プログラムの先頭ラインから、すべてのプログラムを書き出します。

ＬＬＩＳＴ式$_1$ **ENTER**

式$_1$の値で示されるラインの内容のみを書き出します。

ＬＬＩＳＴ式$_1$， **ENTER**

式$_1$の値で示されるラインから、最後のラインまでのプログラムを書き出します。

ＬＬＩＳＴ，式$_2$ **ENTER**

プログラムの先頭ラインから、式$_2$の値で示されるラインまでのプログラムを書き出します。

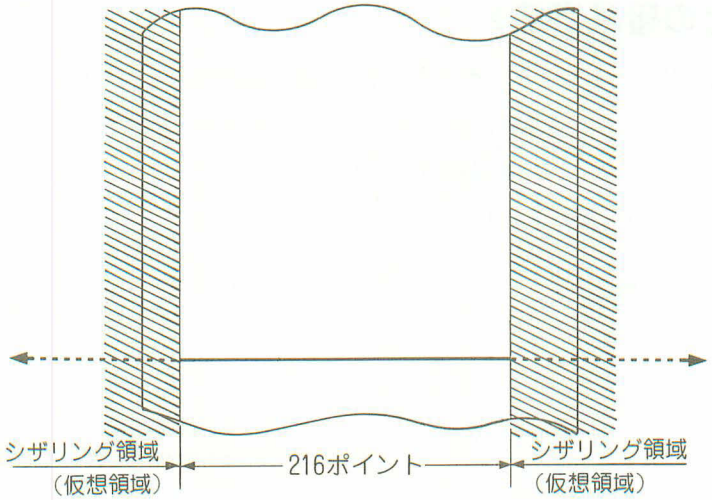
○リストされる文字の大きさはＣＳＩＺＥ　１　あるいはＣＳＩＺＥ　２に限定されます。
ＣＳＩＺＥ　２　以上の文字サイズを指定したときは、すべてＣＳＩＺＥ　２　で印字され、
ＣＳＩＺＥ　１　を指定したときはそのサイズで印字されます。

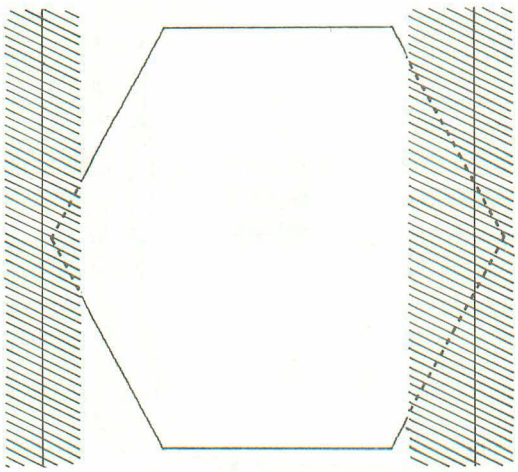
(注)　ＭＥＲＧＥ命令(テープレコーダーに関する命令：　98ページ参照)により、複数のプログラムが同時に計算機内に記憶されているときは、ＬＬＩＳＴに続く式$_1$、式$_2$は最初のプログラムのラインを指定します。したがって、たとえば、ＬＬＩＳＴ50，**ENTER** と押した場合は、最初のプログラムの50ラインから最後のプログラムの最終ラインまでがリストアウトされます。

本プリンタで図形を書く場合は、できるだけ上下方向のくり返し(前後の紙送り)を少なくしたほうが、より正確な図形が描けます。
上下方向にくり返しが多い図形を書いたときは、紙送りの誤差が累積されて、図形のずれが大きくなることがあります。
したがって図形をプログラミングするときは上下方向のくり返しの少ないプログラムを用いてください。

＝ペンおよび紙の移動範囲について＝

本プリンタに使用する紙の幅は約58mmですので、図形を書く場合などはその大きさに制限があります。しかしながら、図形の一部のみを書き出したいが、全体を書き出すプログラムを作るほうが簡単になる場合や、全体を紙幅分ずつに分割してでも大きく書きたい場合などがあります。このようなときに便利な機能がシザリング機能です。この機能は、紙の上にあたる部分の図形は実際に書き、紙の上を外れる部分は、その部分にペンが移動できると仮定して、仮想的に図形を書く機能です。





```
10:GRAPH
20:GLCURSOR (130,
   -200):SORGN
30:C=150,D=0
40:FOR J=0TO 360
   STEP 60
50:A=150*COS J
60:B=150*SIN J
70:LINE (C,D)-(A,
   B)
80:C=A,D=B
90:NEXT J
100:END
```

また、逆方向に紙を送った場合に10.24cmまでしか送られませんが、それ以上になる値をＹ軸方向に指定して図形を書く場合も、シザリング機能が働きます。

● シザリング機能は便利な機能ですが、プログラムを誤りますと仮想領域にすべての図形を書いてしまい、紙上にはなにも書かないことにもなりますので、プログラムを作るには十分注意が必要です。

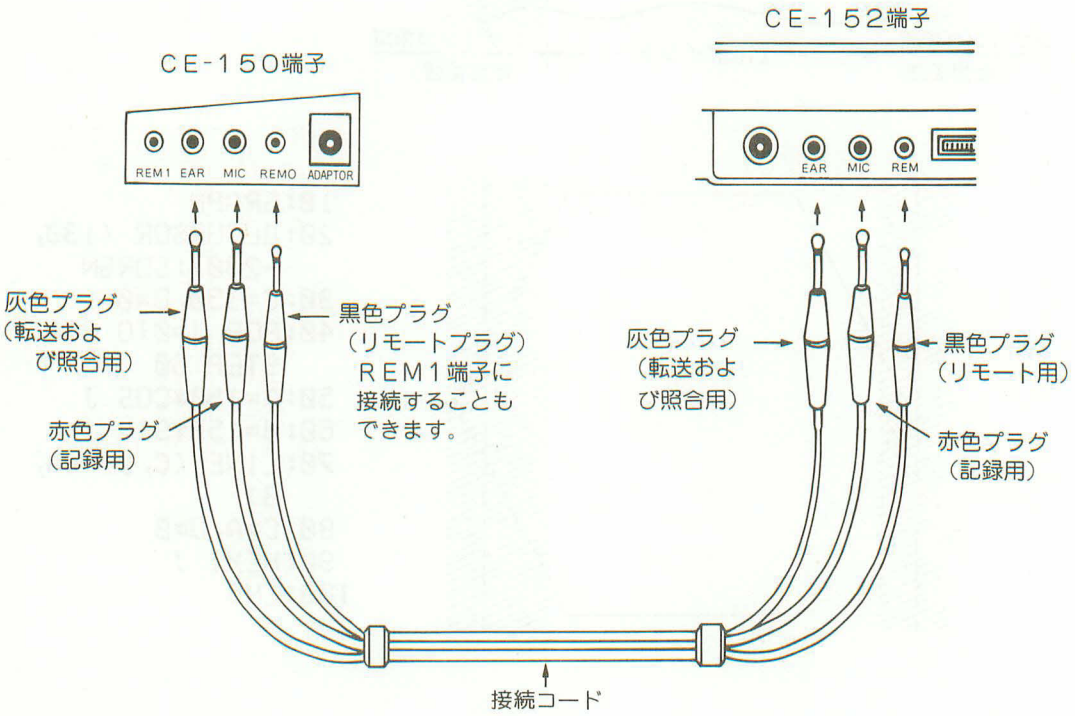
＝逆紙送りカウンターについて＝

本プリンタには、逆紙送りにおいて、10.24cm以上紙を逆送りできないようになっています。この長さは常にカウンターによりカウントされていますので、図形を書くときの原点などとは関係なく、１番前まで送られた位置から逆方向に10.24cmまで送られる(もどされる)と、逆紙送りは止まります。(ＧＲＡＰＨモードではシザリング領域に入ります。)

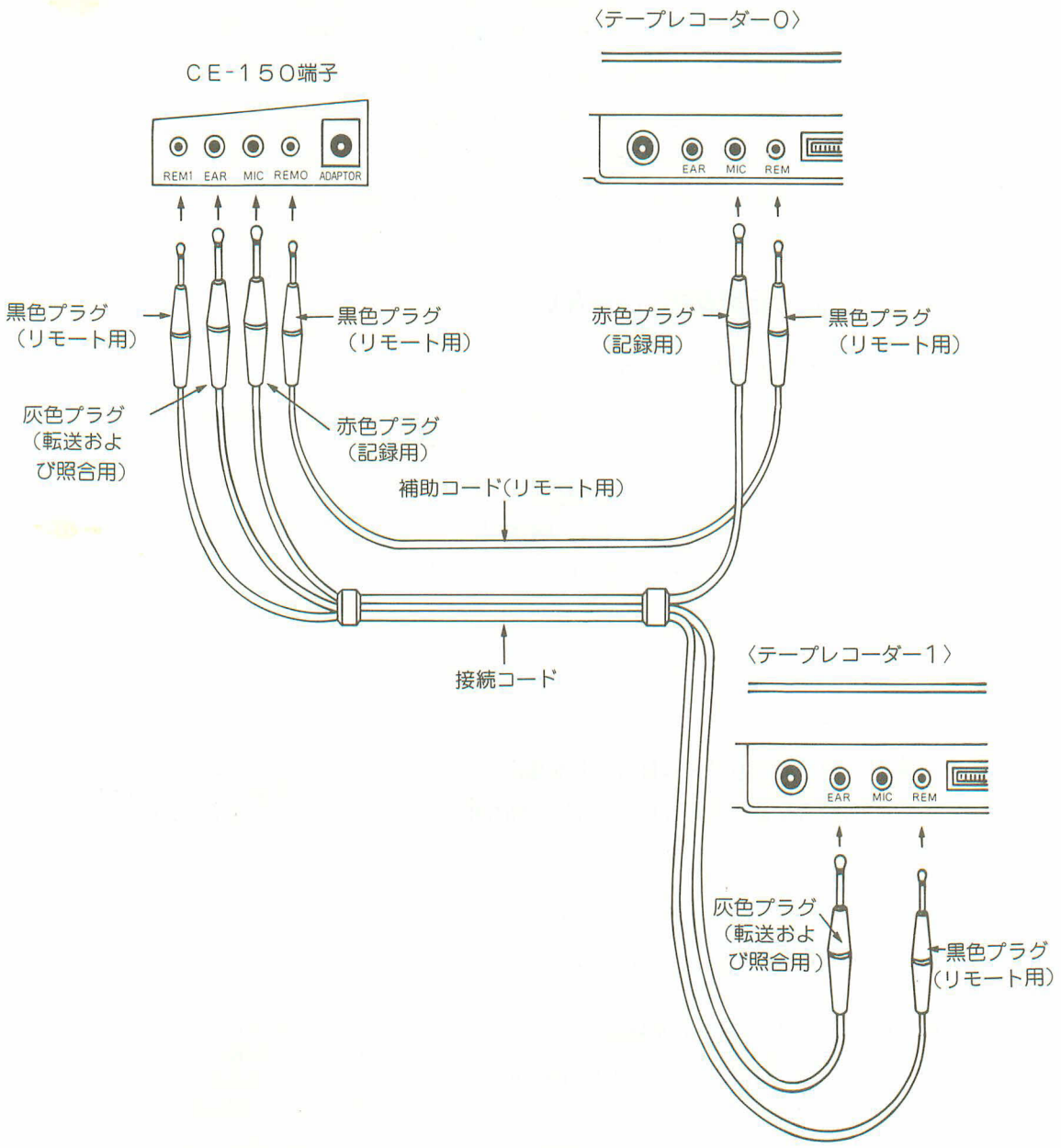
〈テープレコーダーとの接続方法〉

カラーグラフィックプリンタCE-150(カセットインターフェイス内蔵)にはテープレコーダーを１台あるいは２台接続して使用することができます。(テープレコーダーはシャープポケットコンピュータ専用カセットテープレコーダーＣＥ-152をご使用ください。)
ここではＣＥ－152との接続方法について説明します。

① テープレコーダー１台を接続する場合



② テープレコーダー２台を接続する場合



テープレコーダーを２台使用する場合は、どちらか一方を記録用とし、もう一方を読み込み用(転送用)とします。そして図のように接続コードと、リモート用補助コードとを用いて接続します。

- 計算機からプログラムあるいはマニュアル操作によりテープレコーダーの回転を止めたり、動かしたりさせるときにリモート端子を使いますが、ＣＥ-150にはＲＥＭ　０とＲＥＭ　１の２つのリモート端子があります。この端子はどちらを使ってもかまいませんが、プログラムなどでは、その命令の中でＲＥＭ　０端子に接続されているテープレコーダーか、あるいはＲＥＭ　１に接続されているテープレコーダーかを指定しますので、プログラムなどに合わせて接続する必要があります。

テープレコーダーに関する命令

カセットテープなどにプログラムやデータを記録したり、計算機に読み込んだり（転送したり）、あるいは記録されている内容を照合したりする命令について説明します。なお、CE－150にはテープレコーダーを２台接続することができますので、説明の中でテープレコーダーを指定するとき、必要に応じて次のように記載します。

テープレコーダー０：CE-150のREM　０端子に接続しているテープレコーダー
テープレコーダー１：CE-150のREM　１端子に接続しているテープレコーダー

（１）プログラムの記録命令（CSAVE）

◎ CSAVE命令（カセット・セイブ）

プログラムやリザーブ内容をテープに記録する（セイブする）命令で、次のような形で指定します。

① CSAVE“ファイル名” ENTER
② CSAVE－１“ファイル名” ENTER

①の形はテープレコーダー０が指定され、②の形ではテープレコーダー１が指定されます。

○プログラムの記録

プログラムをテープに記録する場合はPROモードかRUNモードでCSAVE（CSAVE－１）命令を実行します。

（例） PROモードかRUNモードを指定
CSAVE "PROG. －1" ENTER

ファイル名として「PROG.－1」をテープに記録したのち、計算機内のプログラムをすべて記録します。

○リザーブ内容の記録

リザーブ内容をテープに記録する場合はRESERVEモードでCSAVE（CSAVE－１）命令を実行します。

（例） RESERVEモードを指定
CSAVE "RESRV－1" ENTER

ファイル名として「RESRV－１」をテープに記録したのち、リザーブ内容とキーシンボル内容をすべて記録します。

記録をおこなったあと、プログラムなどが確実に記録されたことを確認するために、あとで説明しますCLOAD？命令を実行してください。

注）●同じテープの同じ面（A面あるいはB面）に、同じファイル名で内容の違うプログラム、あるいはリザーブ内容を記録することは避けてください。同じファイル名のものが２つ以上ありますと、テープから読み込む（転送する）ときにまちがった内容を読み込む恐れがあります。
●１本のテープに複数のプログラムなどを記録するときは、プログラムとプログラムが重ならないように少し間をあけて記録してください。

記録をおこなうとき、テープレコーダーのテープカウンターの数字を紙などに控えておけば、転送などをおこなうとき、記録した部分を簡単に探すことができます。

○ファイル名

ファイル名はプログラムなどにつける名前(見出し)のことで、ファイル名をつけてプログラムなどを記録しておけば、あとで転送などをおこなうとき、計算機によりファイル名を探し出して転送などをおこなうことができますので便利です。なお、ファイル名を省略することもできます。

(2)プログラムの転送命令(ＣＬＯＡＤ，ＭＥＲＧＥ，ＣＨＡＩＮ)

転送するときはテープを巻きもどし、あるいは送って、転送する内容が記録されている部分の直前までテープを進めてください。

◎ ＣＬＯＡＤ命令(カセット・ロード)

テープに記録されているプログラムやリザーブ内容を計算機に転送するための命令で、次のような形で指定します。なお、この命令は**マニュアル操作でのみ実行**できます。

① ＣＬＯＡＤ “ファイル名” ENTER
　あるいは　ＣＬＯＡＤ ENTER
　（テープレコーダー０が指定されます。）

② ＣＬＯＡＤ－１ “ファイル名” ENTER
　あるいは　ＣＬＯＡＤ－１ ENTER
　（テープレコーダー１が指定されます。）

○プログラムの転送

テープに記録されているプログラムを計算機に転送する場合はＰＲＯモードかＲＵＮモードでＣＬＯＡＤ命令を実行します。

（例） ＰＲＯモードかＲＵＮモードを指定
　ＣＬＯＡＤ　"ＰＲＯＧ．－１" ENTER
　ファイル名が「PROG.－1」のプログラムを検索して、これを計算機のプログラムエリアに転送します。

ファイル名をつけずにＣＬＯＡＤ ENTER と操作した場合はテープレコーダーが動き始めて最初に記録されているプログラムを計算機に転送します。

○リザーブ内容の転送

テープに記録されているリザーブ内容を計算機に転送させる場合はＲＥＳＥＲＶＥモードでＣＬＯＡＤ命令を実行します。

（例） ＲＥＳＥＲＶＥモードを指定
　ＣＬＯＡＤ　"ＲＥＳＲＶ－１" ENTER
　ファイル名が「ＲＥＳＲＶ－１」のリザーブ内容を検索して、これを計算機のリザーブメモリーに転送します。

ファイル名をつけずにＣＬＯＡＤ ENTER と操作した場合はテープレコーダーが動き始めて最初に記録されている内容を計算機に転送します。

注１）モードの指定をまちがえて転送をおこないますとエラーになります。

注2）指定されたファイル名が検索できなかったときは、テープが回り終っても計算機はファイル名を検索し続けます。この場合は BREAK ON キーを押して検索を止めてください。
これは、のちほど説明しますMERGE、CHAIN、CLOAD?、およびINPUT#命令でも同じです。

注3）CLOAD命令およびのちほど説明のCHAIN命令を実行中にエラーが発生しますと、計算機内のプログラムは無効になります。

◎ MERGE命令（マージ）

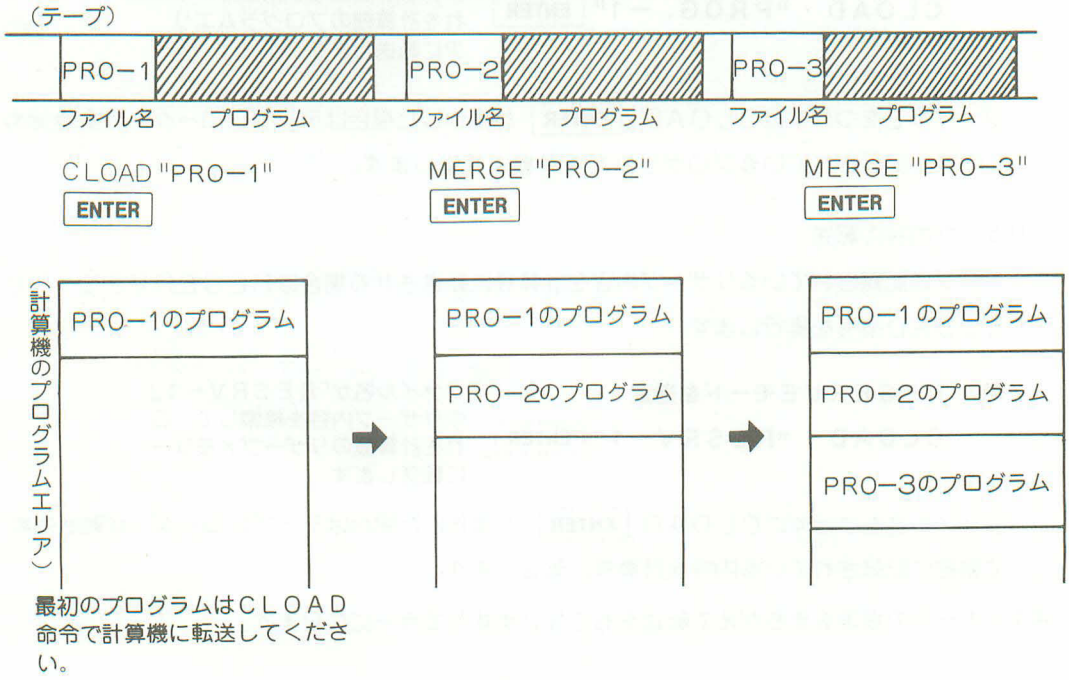
MERGE命令はテープに記録されているプログラムを計算機に転送する命令で、CLOAD命令と同じ形で指定し、実行します。（マニュアル操作でのみ実行できます。）

① MERGE “ファイル名” ENTER
あるいは MERGE ENTER
（テープレコーダー0が指定されます。）

② MERGE－1 “ファイル名” ENTER
あるいは MERGE－1 ENTER
（テープレコーダー1が指定されます。）

CLOAD命令とMERGE命令の違いは、CLOAD命令が前に計算機に記憶されていたプログラムをすべて消してからテープに記録されているプログラムを読み込んでくるのに対して、MERGE命令は前のプログラムを保持したまま、それらの後に、テープに記録されている内容を追加していく形で読み込みます。したがって、全然別のプログラムを計算機に同時に記憶させておくことができます。

（例） テープにPRO－1～PRO－3のファイル名を持つプログラムが記録されているとき、PRO－1をCLOAD命令で、PRO－2、PRO－3をMERGE命令で計算機に転送した場合の記憶の状態は次のようになります。



MERGE命令により読み込まれたプログラムは前例のように記憶されますが、このとき、それぞれのプログラムに使用するラインナンバーやジャンプラベルなどに制約はなく、同じラインナンバーを持つプログラムを同時に記憶させておくことも可能です。

しかしながら、この場合2番目、3番目……に記憶されているプログラムを実行する方法がなくなりますので、これらのプログラムは事前に“定義付けプログラム”として、[A]、[S] ……などのキーに定義付けしておく必要があります。

（定義付けプログラムについては 39 ページを参照ください。）

なお、MERGE命令実行後のプログラム編集は最後に転送し、記憶されているプログラムに対してのみおこなうことができます。

また、これらのプログラム全部を一度テープに記録し、再度計算機に読み込ませた場合、プログラム編集は先頭のプログラムに対してのみおこなうことができます。

注）MERGE命令により、複数のプログラムが記憶されているとき、GOSUB命令でほかのプログラムへサブルーチンジャンプする場合には、そのGOSUB命令に続いてGOTO命令を入れ、実行が復帰してきたときに一度GOSUB命令が書かれているラインより大きいラインへジャンプさせてください。

（例）
```
10 "A": INPUT A
20 GOSUB "B":GOTO 30
30 GOTO 10
10 "B":PRINT A
20 RETURN
```

◎ **CHAIN命令（チェイン）**

CHAIN命令は**プログラムの中で使用する命令**で、テープに記録されているプログラムを計算機に読み込んで、そのプログラムを実行していく命令です。（プログラムでのみ実行できます。）この命令を用いますと、計算機のプログラムエリア内に収まり切れないような長いプログラムでも、何分割かしてテープに記録しておき、次々と読み込みながら実行を続けていくことができます。

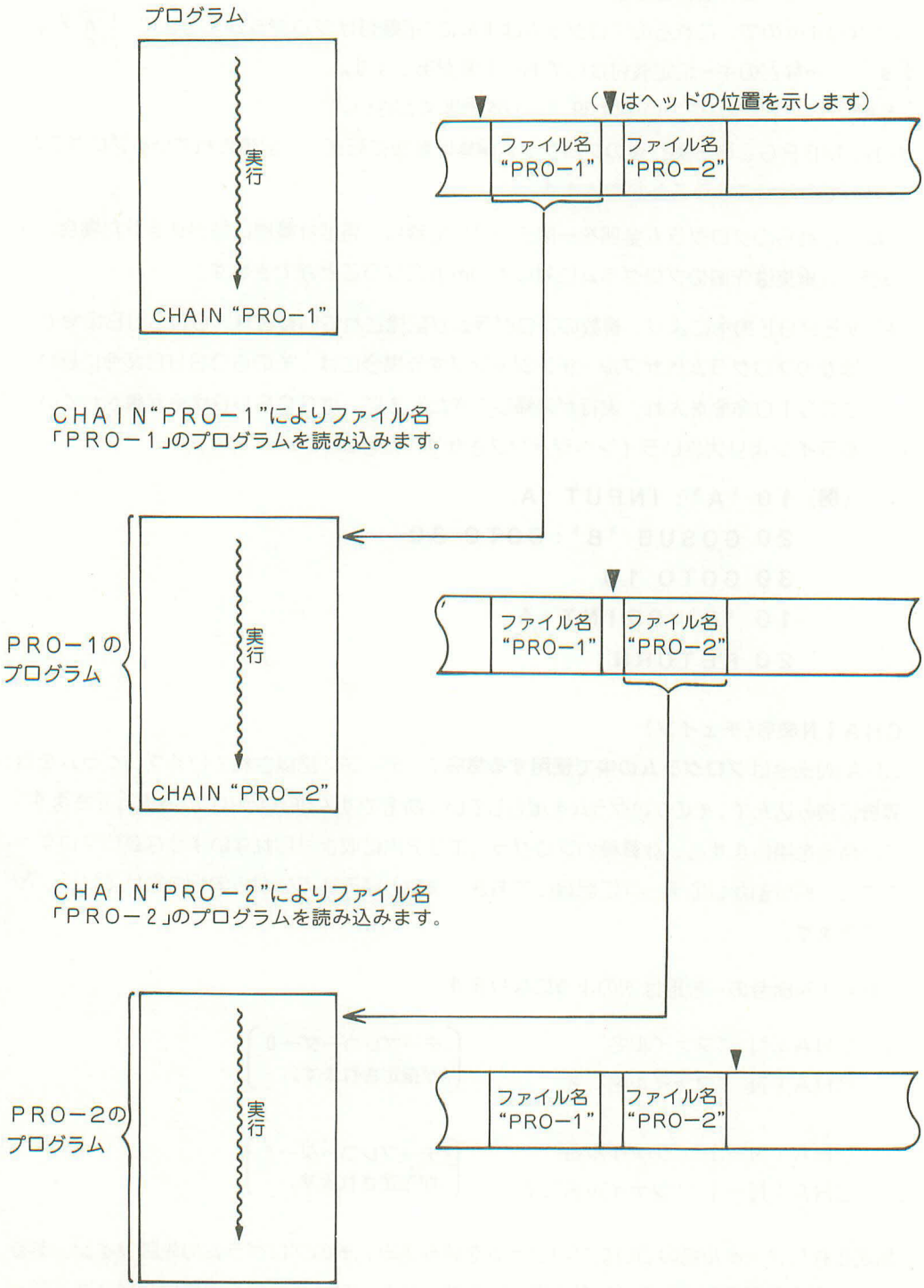
CHAIN命令の一般形は次のようになります。

① CHAIN “ファイル名”
　CHAIN “ファイル名”,式　（テープレコーダー0が指定されます。）

② CHAIN－1 “ファイル名”
　CHAIN－1 “ファイル名”,式　（テープレコーダー1が指定されます。）

指定されたファイル名のついたプログラムを読み込み、そのプログラムの先頭ライン、あるいは式の値で指定されたラインから実行します。なお、ファイル名を省略した場合は、テープが回り出して最初のプログラムを読み込みます。

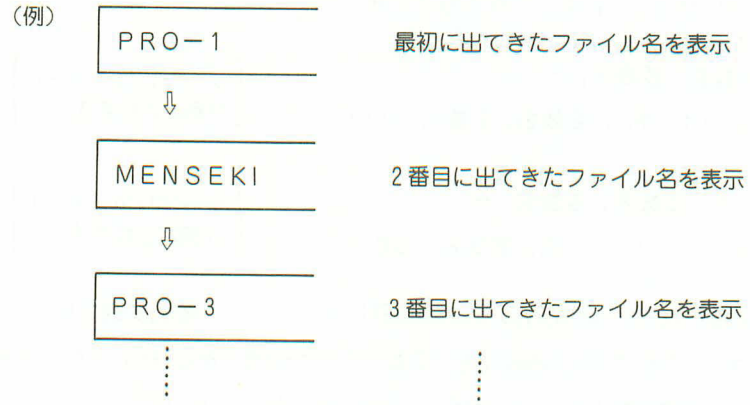
（例） ＣＨＡＩＮ命令により、次のようにプログラムを読み込み、実行します。



このようにプログラムの最後にＣＨＡＩＮ文を入れておくと、それぞれのプログラムの実行の最後に、自動的に磁気テープから次のプログラムを読み込んで実行を続けていくことが可能になります。

＝検索中の表示＝

プログラムなどを検索しているとき、表示部には、途中でテープから読んだファイル名がつぎつぎと表示されます。



（3）照合命令（CLOAD?）

◎ CLOAD?命令（カセット・ロード?）

計算機に記憶させているプログラムあるいはリザーブ内容と、テープに記憶されている内容との照合をおこなう命令で、次のような形で指定します。なお、この命令は**マニュアル操作による実行のみ可能**です。

① CLOAD?“ファイル名” ENTER
あるいは CLOAD? ENTER

② CLOAD?－1 “ファイル名” ENTER
あるいは CLOAD?－1 ENTER

この命令により、もし不一致が生じたときは、エラー（ERROR 43）になります。
不一致がなければプロンプト表示になります。

○プログラムの照合

計算機に記憶されているプログラムとテープに記録されているプログラムを照合する場合はPROモードあるいはRUNモードでCLOAD?命令を実行します。

（例） **PROモードかRUNモードを指定**
CLOAD? "PRO－1" ENTER

ファイル名が「PRO－1」のプログラムを検索し、その内容と、計算機のプログラムとを照合します。

○リザーブ内容の照合

計算機に記憶されているリザーブ内容とテープに記録されているリザーブ内容を照合する場合はRESERVEモードでCLOAD?命令を実行します。

（例） **RESERVE モードを指定**
CLOAD? "RESV－1" ENTER

ファイル名が「RESV－1」のリザーブ内容を検索し、その内容と、計算機のリザーブ内容とを照合します。

（4）データの記録命令（ＰＲＩＮＴ＃）

◎ ＰＲＩＮＴ＃命令（プリント・クロスハッチ）

計算機内に記憶されているデータをテープに記録する命令で、次のような形で指定します。（マニュアル操作、プログラムとも実行できる命令です。）

① ＰＲＩＮＴ＃変数名，変数名，……
ＰＲＩＮＴ＃“ファイル名”；変数名，変数名，……
（テープレコーダー０が指定されます）

② ＰＲＩＮＴ＃－１，変数名，変数名，…
ＰＲＩＮＴ＃－１，“ファイル名”；変数名，変数名，…
（テープレコーダー１が指定されます。）

指定された変数に記憶されている内容をテープに記録します。なお、配列変数の場合は（　）の中に＊を指定します。この形で、同じ名前の配列変数すべてを同時に指定することができます。配列変数（数値変数、文字変数とも）はこの形でのみ記録することが可能です。

（例） ＰＲＩＮＴ＃Ａ，Ｂ，ＡＡ
変数Ａ，Ｂ，ＡＡの内容をファイル名なしでテープに記録します。

ＰＲＩＮＴ＃"ＤＡＴＡ－１"；Ｂ（＊）
ファイル名「ＤＡＴＡ－１」をつけて、Ｂ（　）の変数（Ｂ（０）、Ｂ（１）、……）をテープに記録します。

（注）ＡＡやＢ１＄のような２文字の変数はＰＲＩＮＴ＃命令を実行する前に、プログラム・データエリアに確保されていなければなりません。
もし確保されていない場合には計算機が、その変数名を探し続け（約30秒間）、その後、変数名が正しくないものと見なしてエラー（ＥＲＲＯＲ　21）になります。

（注）ＰＲＩＮＴ＃変数名，変数名，……という形（ファイル名がない形）において、最初の変数が文字変数の場合は、その文字変数の内容がファイル名とみなされてエラー（ＥＲＲＯＲ　１）になります。これはＩＮＰＵＴ＃命令においても同じです。

（5）データの転送命令（ＩＮＰＵＴ＃）

◎ ＩＮＰＵＴ＃命令（インプット・クロスハッチ）

テープに記録されているデータを指定された変数に転送する命令で、次のような形で指定します。（マニュアル操作、プログラムとも実行できる命令です。）

① ＩＮＰＵＴ＃変数名，変数名，…
ＩＮＰＵＴ＃“ファイル名”；変数名，変数名，…

② ＩＮＰＵＴ＃－１，変数名，変数名，…
ＩＮＰＵＴ＃－１，“ファイル名”；変数名，変数名，…

テープに記録されているデータを指定された変数に転送します。なお、配列変数の場合は、ＩＮＰＵＴ＃命令を実行する前にＤＩＭ命令で変数名と大きさを定義しなければなりません。なお、その配列の大きさ、長さなどはテープに記録されている内容の配列の大きさ、長さなどと一致していなければなりません。

（例）　ＩＮＰＵＴ＃Ａ，Ｂ，ＡＡ　　テープが回りはじめて最初に記録されているデータを変数Ａ，Ｂ，ＡＡに転送していきます。

```
    ⋮
50  DIM  B(5)
60  INPUT# "DATA−1" ;B(*)
    ⋮
```

ファイル名が「ＤＡＴＡ−１」のデータを検索して、変数Ｂ（０）～Ｂ（５）に転送していきます。

注）ＩＮＰＵＴ＃命令で指定した変数の数とテープに記録されているデータの数が違っている場合は次のようになります。

- ○指定した変数の数よりテープに記録されているデータの数が多い場合は、指定した変数にデータを転送し、残ったデータは無視します。
- ○指定した変数の数よりテープに記録されているデータの数が少ない場合は、記録されているデータが指定されている変数に順番に転送された後、エラー（ＥＲＲＯＲ　４３）になります。
- ○ＩＮＰＵＴ＃命令に続いて変数を指定していない場合はエラー（ＥＲＲＯＲ　４３）になりります。

（６）リモートコントロール機能について（ＲＭＴ　ＯＦＦ，ＲＭＴ　ＯＮ）

通常、ＣＥ-150のリモート端子（ＲＥＭ０、あるいはＲＥＭ１）とテープレコーダーのリモート端子（ＲＥＭ）とを接続した場合、テープレコーダーの再生ボタン（▶）を押してもテープは回らず、計算機からの記録命令や転送命令などが実行されたとき、リモート機能が働いてテープが回ります。そして、記録や転送などが終了すればテープが止まります。（記録、転送、照合の各命令の−１がついた形のとき、ＣＥ-150のＲＥＭ　１端子にコントロール信号が出され、−１がつかない形の命令ではＲＥＭ　０端子にコントロール信号が出されます。）

◎ **リモート機能の解除**

プログラムの頭出し(プログラムの頭の部分を探し出すこと)をおこなうときや、記録されていない(空いている)部分を探すときなどはリモート機能を解除してからでないとテープが回りません。

○リモート機能の解除は次の方法でおこないます。

REM　0端子：プリンタのリモートスイッチをOFF側にします。

テープレコーダー0はすべて手動で動作します。

REM　1端子：RMT　OFF命令(リモート・オフ)を実行します。

RMT　OFF [ENTER]　テープレコーダー1はすべて手動で動作します。

○解除したリモート機能をふたたびセットするには

REM　0端子：プリンタのリモートスイッチをON側にします。

テープレコーダー0はリモート機能によりコントロールされます。

REM　1端子：RMT　ON命令(リモート・オン)を実行します。

RMT　ON [ENTER]　テープレコーダー1はリモート機能によりコントロールされます。

計算機とテープレコーダーの操作方法

プログラムなどをテープに記録する、あるいはテープから転送する場合の計算機とテープレコーダーの操作方法について説明します。

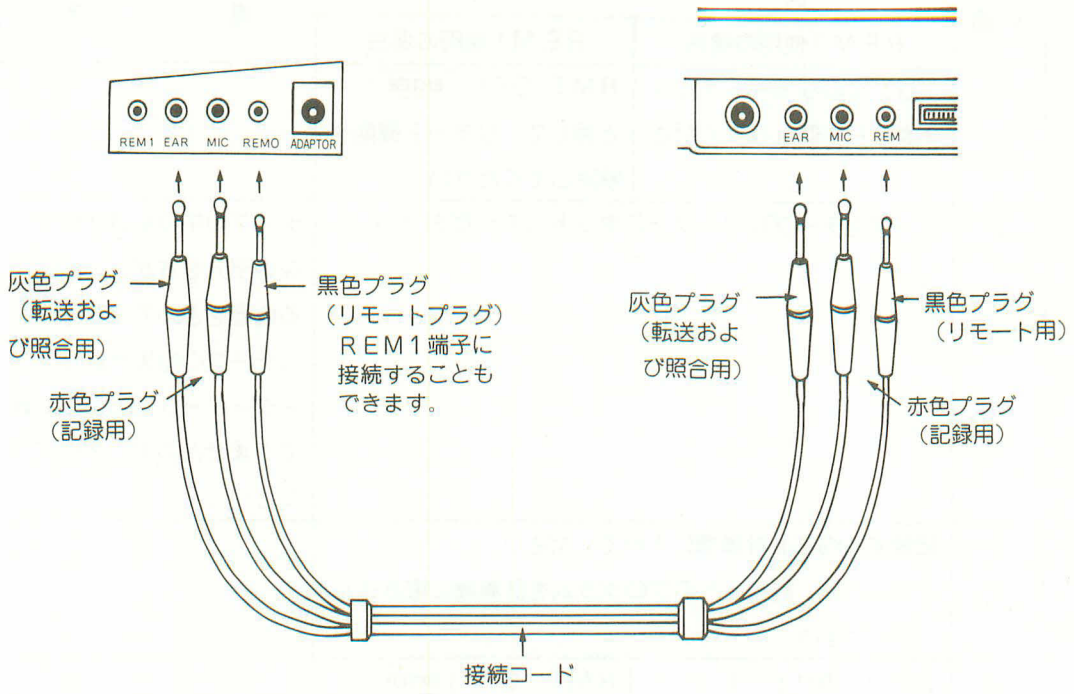
注）テープレコーダーは別売のポケットコンピュータ専用カセットテープレコーダーCE-152を使用されているものとして説明します。

(1) 準備

◎ **各機器の接続**

○計算機の [OFF] キーを押して電源をオフにし、計算機とCE-150を接続してください。(接続方法は 82 ページ参照)

○CE-150とCE-152を次のように接続してください。(テープレコーダーを1台使用する場合。)



●ＣＥ-152の取扱説明書も合わせてお読みください。

◎ **テープレコーダーに関する注意**

●転送あるいは照合をおこなうときは、記録をおこなったテープレコーダーを使用してください。記録したテープレコーダーと転送あるいは照合するテープレコーダーが異なる場合は転送あるいは照合できないことがあります。

●テープレコーダーのヘッドなどが汚れていますと、ひずみの増加、レベルの低下などがおこりますので、ヘッドの清掃をおこなってください。

●テープは一般に市販のものをご使用ください。ただし、極端に周波数特性の悪いもの、また、傷や折目があるものは使用しないでください。

(2)テープへの記録方法

例としてプログラムを記録する方法について説明します。〈マニュアル操作の場合〉

〈プログラム〉

```
10  INPUT "HANKEI_R=";A
20  S=π*A*A
30  PRINT "MENSEKI_S=";S
40  END
```

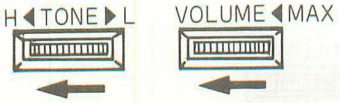
手順	操作		備考
	REM0使用の場合	REM1使用の場合	
(1)	プリンタのリモートスイッチをOFF側にしてください。	RMT OFF ENTER と押して、リモート機能を解除してください。	
(2)	テープをテープレコーダーにセットしてください。		テープの中の記録されていない部分をさがし、記録する位置をきめてください。（テープの始めや終りのリーダーテープ部分には記録できませんので、さけてください。）
(3)	記録する内容を計算機に入れてください。 たとえば、前ページのプログラムを計算機に書き込んでください。		
(4)	プリンタのリモートスイッチを、ON側にしてください。	RMT ON ENTER と押して、リモート機能をセットしてください。	
(5)	テープレコーダーの記録ボタン(●)を押し込んで、録音状態にしてください。		
(6)	記録命令を実行します。 プログラム： CSAVE “ファイル名” ENTER データ： PRINT #“ファイル名”；変数，変数，…… ENTER (例)PROモードかRUNモードを指定 CSAVE"PR-1" ENTER	記録命令を実行します。 プログラム： CSAVE-1“ファイル名” ENTER データ： PRINT#-1,“ファイル名”；変数，変数，…… ENTER (例)PROモードかRUNモードを指定 CSAVE-1"PR-1" ENTER	記録命令が実行されますと、まず最初に約 8 秒間無信号区間が記録（ピー音が記録）され、そのあとにファイル名とプログラムなどが記録されます。（プログラムなどの記録中は ピピピピ……あるいは ビビビビ …というような音が出ます。）
	記録が終了しますと、プロンプト表示になり、テープが止まります。 テープを巻きもどして、照合をおこなってください。 巻きもどし、早送りなどをおこなうときは、プリンタのリモートスイッチをOFF側にして、おこなってください。	巻きもどし、早送りなどをおこなうときは、RMT OFF ENTER と操作して、リモート機能を解除してからおこなってください。	照合については、101ページを参照ください。

● プログラムの実行において、データの記録をおこなう場合（ＰＲＩＮＴ＃ 命令）には、プログラム実行前にテープレコーダーの操作をしておいてください。

記録をおこなうとき、記録を始めたときのテープカウンタの数値を紙などに控えておけば、記録位置や、記録されていない位置を探し出すときに便利です。

（3）計算機とテープの内容の照合

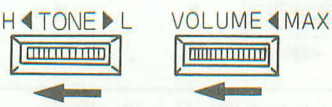
計算機に記憶されているプログラムあるいはリザーブ内容と、テープに記録されているプログラムあるいはリザーブ内容との照合を取る場合（マニュアル操作でのみ実行可能）には次のようにおこないます。

手 順	操作		備 考
	ＲＥＭ０使用の場合	ＲＥＭ１使用の場合	
(1)	プリンタのリモートスイッチをOFF側にしてください。	ＲＭＴ ＯＦＦ ENTER と押して、リモート機能を解除してください。	
(2)	照合する内容が記録されているテープをテープレコーダーにセットしてください。 そして、テープカウンターなどを見ながら、照合する内容の記録されている位置の近くまでテープを送ってください。		
(3)	プリンタのリモートスイッチをＯＮ側にしてください。	ＲＭＴ ＯＮ ENTER と押して、リモート機能をセットしてください。	
(4)	音量調整つまみ、音質調整つまみを図の矢印方向いっぱいに回しておいてください。 		
(5)	テープレコーダーの再生ボタン（▶）を押し込んで、再生状態にしてください。		
(6)	照合命令を実行します。 ＣＬＯＡＤ？“ファイル名” ENTER (例)ＰＲＯモードかＲＵＮモードを指定 ＣＬＯＡＤ？ "ＰＲ-1" ENTER	照合命令を実行します。 ＣＬＯＡＤ?-1“ファイル名” ENTER (例)ＰＲＯモードかＲＵＮモードを指定 ＣＬＯＡＤ?-1"ＰＲ-1" ENTER	ファイル名は記録をしたファイル名を入れてください。ファイル名を省略した場合はテープが回り出して最初の記録内容と、計算機内の内容との照合をおこないます。
	両方の内容が一致している場合は実行を終了しプロンプト表示になります。		

	もし、エラー表示（ＥＲＲＯＲ 43）になった場合は、もう一度最初から照合をおこなってください。 それでもなお、エラーになる場合は、もう一度“記録”からおこなってください。	ファイル名やモード指定などにまちがいがないか、よく確認しながら操作してください。

（４）テープからの転送

テープに記録されているプログラムあるいはリザーブ内容、データを計算機に転送する（テープより読み出す）場合は次のようにおこなってください。（マニュアル操作の場合）

手 順	操作		備 考
	ＲＥＭ 0 使用の場合	ＲＥＭ 1 使用の場合	
(1)	プリンタのリモートスイッチをOFF側にしてください。	ＲＭＴ ＯＦＦ [ENTER] と押して、リモート機能を解除してください。	
(2)	転送する内容が記録されているテープをテープレコーダーにセットしてください。 そして、テープカウンターなどを見ながら、転送する内容の記録されている位置の近くまでテープを送ってください。		
(3)	プリンタのリモートスイッチをＯＮ側にしてください。	ＲＭＴ ＯＮ [ENTER] と押して、リモート機能をセットしてください。	
(4)	音量調整つまみ、音質調整つまみを図の矢印方向いっぱいに回しておいてください。 		
(5)	テープレコーダーの再生ボタン（▶）を押し込んで、再生状態にしてください。		
(6)	転送命令を実行します。 プログラム： ＣＬＯＡＤ“ファイル名” [ENTER] データ： ＩＮＰＵＴ #“ファイル名”；変数，変数，… [ENTER]	転送命令を実行します。 プログラム： CLOAD-1 “ファイル名” [ENTER] データ： INPUT #-1,“ファイル名”；変数，変数，… [ENTER]	ファイル名は記録をしたファイル名を入れてください。 ファイル名を省略した場合はテープが回り出して最初に記録されている内容が転送されます。

	(例)ＰＲＯモードかＲＵＮモードを指定 ＣＬＯＡＤ "ＰＲ-1" ENTER	(例)ＰＲＯモードかＲＵＮモードを指定 ＣＬＯＡＤ-1 "ＰＲ-1" ENTER	
	転送が終了すればプロンプト表示になります。		
	もし、途中でエラー（ＥＲＲＯＲ 43、ＥＲＲＯＲ 44）になった場合は、もう一度最初から転送をおこなってください。 それでもなお、エラーになる場合は、もう一度“記録”からおこなってください。		ファイル名やモード指定などにまちがいがないか、よく確認しながら操作してください。

- プログラムの実行において、転送をおこなう場合（ＣＨＡＩＮ命令、ＩＮＰＵＴ＃命令）には、プログラム実行前にテープレコーダーの操作をしておいてください。

〈その他の注意〉

① 各機器の接続は確実におこない、連動中には抜かないでください。着脱は必ず計算機の電源をオフにしてからおこなってください。

② ＡＣ電源や、ＡＣアダプタでテープレコーダーを使用して、連動がうまくいかないとき、あるいは計算機が正常に動作しないときには、電池電源を使用してください。
ＡＣラインノイズのひどい環境で、ＡＣ電源でテープレコーダーを使用しますと、たとえば連動がうまくいかない、計算機が正常な動作をおこなわない（誤動作をする）などの現象が表われることがあります。（このようなとき、市販の雑音防止器を使用すると効果がある場合があります。）

（5）CE-152以外のテープレコーダーを使用する場合について

ＣＥ-152以外のテープレコーダーでも次のような条件を満足すれば使える場合があります。

条　　項	条　　件
種　　類	カセットテープ、マイクロカセットテープ、オープンリールテープを使用するテープレコーダーおよびテープデッキ
入力端子 ●入力インピーダンス ●最小入力レベル	ＭＩＣ（マイク）端子あるいはそれに代る端子を有すること（ミニジャック） 注）ＡＵＸ（補助入力）端子は使用不可 上記端子が、インピーダンス200Ω～１ＫΩ以下もしくはローインピーダンス用であること ３ｍＶ以下もしくは－50ｄＢ以下
出力端子 ●出力インピーダンス ●出力レベル	ＥＸＴ．ＳＰＥＡＫＥＲ（外部スピーカー）端子あるいはそれに代るＭＯＮＩＴＯＲ（モニター）端子、ＥＡＲＰＨＯＮＥ（イアホーン）端子などを有すること（ミニジャック） 上記端子のインピーダンスが10Ω以下であること １Ｖ以上（実用最大出力 100ｍＷ以上であること）
歪率（位相歪などを含む）	15%以内（同位相かつ２ｋＨｚ～４ｋＨｚ内において）
ワウフラッター	0.3%以内（Ｗ.Ｒ.Ｍ.Ｓ.） 注）ワウフラッターはテープレコーダーに振動を与えたり、ショックを与えたりしますと、大きくなります。
そ の 他	極端な回転の変動がないこと

注）テープレコーダーの中には、接続条件が一部合わないもの、あるいは長期使用などにより、歪やノイズの増加、出力の低下など、電気的特性が変化して正常に動作しないことがあります。

キーの機能

次に主なキーの機能を説明します。

キー	機能
BREAK ON	○電源を入れるときに押します。 ○プログラム実行中、プログラムを一時停止状態（ＢＲＥＡＫ状態……ブレーク）にします。 ○表示などをクリヤします。（エラー状態を解除します。）
OFF	○電源を切るときに押します。
SHIFT	○π、∧、？のように、キー上に直接記されずに、薄茶色でパネル上に記されている命令（機能）を指定します。 （例） SHIFT ? ／ → ？ が入力されます。 ○アルファベットの小文字を入力するとき、アルファベットキーの前に押します。
CL	○入力中の内容や表示をクリアするときに押します。（エラー状態を解除します。）
SHIFT CA	○計算機の状態を解除します。（クリアオール） ●プログラムの実行が一時停止状態にあるとき、実行を中止させます。 ●ＷＡＩＴ命令による時間指定を解除します。 ●表示フォーマット指定（ＵＳＩＮＧ指定）を解除します。 ●トレースモードの解除 ●エラー状態の解除 その他
MODE	○ＲＵＮモードとＰＲＯモードとの指定を切替えます。
SHIFT MODE	○ＲＵＮモードとＲＥＳＥＲＶＥモードとの指定を切替えます。
0 ～ 9	○数値を指定します。
.	○小数点を指定します。 ○命令入力の際の省略形の指定をおこないます。 ○ＵＳＩＮＧ命令内において、数値データの表示に小数点以下の表示をおこなう指定をします。
E	○指数部入力を指定します。（このキーはアルファベットキーと同じです。）
／	○除算命令を指定します。
*	○乗算命令を指定します。 ○ＵＳＩＮＧ命令内でも使用します。
+	○数値の符号（正符号）を指定します。（一般に省略可） ○加算命令を指定します。

キー	機能
[−]	○数値の符号（負符号）を指定します。 ○減算命令を指定します。
[SHIFT] [?]	○ＣＬＯＡＤ？命令を指定するときに使用します。
[SHIFT] [:]	○１ラインに複数のステートメントを定義するときの区切りを指定します。
[SHIFT] [,]	○式と式の区切りや変数の区切り、コメントの区切りなどを指定します。 ○ＵＳＩＮＧ命令内で、数値の３桁区切りマークをつけて表示をおこなう指定をします。
[SHIFT] [;]	○式や変数の区切り、コメントの区切り、命令文と変数の区切りなどを指定します。
[=]	○代入文において、右辺の内容を左辺で指定される変数に代入することを指定します。 ○ＩＦ文の条件式などに使用します。
[SHIFT] [@]	○固定変数を添字付変数として指定するときに使用します。 （例）　@（１）、@$（２）　など ○ソフトキーを定義付けキーとして使用する場合などに、リザーブ内容として使用します。 （例）　ＧＯＴＯ　100@　など
[DEF]	○Ａ、Ｓ、Ｄ、…などのラベルで定義されたプログラムの実行を開始するとき、そのラベルに対するキーを押す前に押します。（Definable キー指定） ○Ｑ、Ｗ、Ｅ、…などのキーに定義されている命令を入力するとき、これらのキーの前に押します。
[A] ～ [Z]	○アルファベットキーで、各命令や、変数名などを入力します。 ○ [DEF] キーに続いて押せば定義付けキーとして働きます。 ○ [SHIFT] キーに続いて押せば小文字が入力されます。
[(] 、[)]	○カッコの指定をおこないます。
[SHIFT] [<] [SHIFT] [>]	○ＩＦ文の条件式などに使用します。
[SPACE]	○プログラム入力あるいは文字列入力時にスペースを設けるための命令です。スペースはプログラム、演算などの実行時には無視されます。
[SHIFT] [^]	○べき乗命令を指定します。 ○ＵＳＩＮＧ命令内において、数値データの表示形式を浮動小数点表示（指数表示）形式に指定します。

キー	機能
SHIFT π	○π（パイ）を指定します。
SHIFT √	○開平計算の命令を指定します。
RCL	○表示を、現在の内容と書き込まれているキーシンボル（ソフトキーのキーシンボル）と入れ替えます。
▲▼	○ソフトキーにリザーブされている内容のリザーブナンバー（Ⅰ、Ⅱ、Ⅲ）を切替えます。
カナ	○オプション（別売）のカタカナがあるときに、カナ入力モードの指定と解除をおこないます。
□	○リザーブされている内容を呼び出すとき、あるいはリザーブをおこなうときに押します。（ソフトキー） ○リザーブされていないキーを押したときは、それぞれのキーが持つキャラクターシンボル（！" ＃ ＄ ％ ＆）が入力されます。
SHIFT ! ～ SHIFT &	○それぞれのキャラクターシンボルを入力します。 " ：文字列の指定をおこないます。 ラベルの指定をおこないます。 ＃：ＵＳＩＮＧ命令内において、数値データの表示フォーマットを定義します。 ＰＲＩＮＴ＃命令およびＩＮＰＵＴ＃命令を指定するときに使用します。 ＄：文字変数を指定します。 ＆：ＵＳＩＮＧ命令内において、文字列の表示フォーマットを定義します。 16進数値を示します。 ！ ％ ： 〝 〟 の中で文字列として使用します。
►	○カーソルの右シフトをおこないます。 ○プレイバック命令の実行をおこないます。 ○プログラムおよびリザーブ内容が表示されているとき、カーソルが表示されていない場合にカーソルを呼び出します。（コロンの右隣） ○マニュアル操作時のエラーを解除します。
◄	○カーソルの左シフトをおこないます。 ○そのほか ► キーと同じ。
SHIFT INS	○カーソルが示す位置に、１ステップ容量の挿入スペース（ただし □ 表示）を設けます。
SHIFT DEL	○カーソルが示す位置の内容を削除します。
ENTER	○プログラムのラインの終了命令を指定します。 ○プログラム、リザーブ内容、キーシンボルなどの書き込みをおこないます。 ○マニュアル計算の実行あるいはBASIC命令のマニュアル操作による実行を命令します。 ○ＩＮＰＵＴ命令やＰＲＩＮＴ命令で、一時停止しているプログラムの再スタートを命令します。

[↓] 、[↑] キーについては、モードの指定および計算機の状態により、次のように働きます。

モード	状態	[↓]	[↑]
RUN	プログラム実行中		
	プログラムの一時停止中 INPUT命令実行時 PRINT命令実行時 ブレーク状態	次のラインを実行（１ラインずつ実行して停止します。）	押している間、実行しているラインあるいは実行したラインを表示
	プログラム実行時のエラー状態		押している間、エラー発生のラインを表示
	トレースモードオン状態	デバッグ動作を実行	押している間、実行しているラインあるいは実行したラインを表示
PRO（他のモードからPROモードに切換え、プログラムラインが表示されていないとき）			
	プログラムの一時停止中	実行を停止しているラインを表示	同左
	エラー発生後[ON]あるいは[CL]でエラーを解除したとき。	エラー発生のプログラムラインを表示	同左
	その他	先頭ラインを表示	末尾ラインを表示
PRO（プログラムラインが表示されているとき）			
		次のプログラムラインを表示	１ライン前のラインを表示
RESERVE			

- [ENTER] キーは表示上ではスペースと同じです。
- [OFF] キーはプログラムの実行中など、内部で演算処理をおこなっているときは働きません。
- キー入力待ちの状態で、約７分間キー入力がなければ自動的に電源が切れます。
  （Auto power off：オートパワーオフ機能）

◎ 表示シンボル

表示部の上側に表示されるシンボルの意味を説明します。

BUSY　：プログラムを実行していることを示します。

SHIFT　：[SHIFT] キーが押されたことを示します。このシンボルが表示されているときは、キーの上側のパネルに薄茶色で記されている機能が指定されています。

カナ　：オプションの「カタカナ」を持っている場合に、カナが入力できる状態になっていることを示します。

DEG ：ディグリー〔°〕
RAD ：ラディアン〔rad〕
GRAD ：グラード　〔g〕

三角関数や逆三角関数の計算において、取り扱う角度の単位を示すシンボルです。

$(90° = \frac{\pi}{2}〔rad〕= 100^g)$

RUN ：ランモード
PRO ：プログラムモード
RESERVE ：リザーブモード

計算機のモード（状態）を示すシンボルです。

DEF ：DEF キーが押されたことを示します。このシンボルが表示されているとき、アルファベットキーなどが定義付けキーや割り当てられている命令の入力キーとして働きます。

Ⅰ
Ⅱ
Ⅲ

リザーブナンバー
ソフトウェアキーの何番目の内容かを示します。

● ：バッテリーインジケーター

電源を入れたとき、電池の電圧が十分にある場合は表示部の右はしにグレー（灰色）のバッテリーインジケーター「●」が表示されます。（他のシンボルなどにくらべ、わずかにうすく表示されます。）

電池の電圧が規定より低くなりますと「●」が正面から見えなくなります。

オプション機器のご案内

PC-1501は、次の別売周辺機器を使用することができます。

● CE-150

カラーグラフィックプリンタ（カセットインターフェイス内蔵）

4色のボールペンを用いて、文字はもちろんグラフや図形が描けるプリンタです。

● CE-151

メモリーモジュール（4KバイトRAM）

PC-1501に取りつければ、プログラム・データエリアの容量が10042バイトになります。

● CE-152

ポケットコンピュータ用カセットテープレコーダー

CE-150を介してプログラムやデータをテープに記録、読み込みができます。

● CE-153

ソフトウェアボード

10×14(140)のキーを持つキーボード。

キーボードを集計表などに見立ててプログラムすることにより、販売実績や予算、達成率などの数値を即座に求めるなど、自由な使いかたができます。

● CE-154

システムキャリングケース

PC-1501、CE-150、CE-152、CE-153などの周辺機器を接続したままそっくり収納、持ち運びができるケースです。

● CE-155

メモリーモジュール（8Kバイト RAM）

PC-1501に取りつければ、プログラム・データエリアの容量が14138バイトになります。

● CE-156

カナテープ

カタカナを使用するためのプログラムが記録されたカセットテープです。

● CE-157

カナモジュール（カナROM＋4KバイトRAM）

PC-1501に取りつければ、カタカナが使用できるとともに、プログラム・データエリアが10042バイトになります。

● CE-158

インターフェイス

RS-232Cインターフェイスとパラレルインターフェイスを持つ汎用インターフェイスです。

● CE-159 ・ CE-161

プログラムモジュール（CE-159：8KバイトRAM、CE-161：16KバイトRAM）

モジュールを取り外したときでも、内蔵電池により記憶しているプログラムを保持することのできるモジュールです。

上記周辺機器は、もよりのシャープ製品取扱店でお求めください。

＝CE-159をご使用いただく上でのご注意＝

CE-159内にデータを残す場合は、CE-159取扱説明書52ページに記載のデータ保護用プログラムを用いますが、CE-159をPC-1501で用いてデータを残す場合は、このプログラムの40ライン、50ラインを次のように変更してください。

```
40 "B" : POKE  30873,88,0
50  DIM  ZX$ (255),ZY$ (127),ZZ$ (0) * 2
```

また、51ページに記載のCE-159内で変数として使える大きさを知る操作は次の操作に変えてください。

MEM－（6144＋23） ENTER

機械語について

PC-1501で機械語を使用される場合、190ページに記載の内容に留意していただければ、パソコン販売店などでお求めいただけます「シャープポケットコンピュータPC-1500機械語の手引」がご利用いただけます。
なお、パソコン販売店などでお求めになれない場合はもよりのシャープビジネス(株)の支店・営業所にお問い合わせください。

仕　様

形　　名……ＰＣ-1501

計　算　桁　数……10桁（仮数部）＋２桁（指数部）

計　算　方　式……数式通り（優先順位判別機能つき）

プログラム言語……ＢＡＳＩＣ（ベーシック）

内部システム構成……ＣＰＵ　：　ＣＭＯＳ　８ビットＣＰＵ

ＲＯＭ　：　16K バイト

ＲＡＭ　：　8.5K バイト

システムエリア　1.9K バイト

入力バッファ　　80バイト

スタック　　196バイト

機械語フリーエリア（7C01H～7FFFH）1023バイト

その他

ユーザーエリア　6.6K バイト

データ専用エリア（Ａ～Ｚ、Ａ＄～Ｚ＄）　624バイト

プログラム・データエリア　5946バイト

リザーブエリア　188バイト

基本計算機能……基本計算　：　加減乗除算

関数計算　：　三角関数、逆三角関数、対数、指数、角度変換、べき乗、開平計算、整数化、絶対値、符号関数、円周率

編　集　機　能……カーソル左右シフト（▶，◀）

挿　入（ＩＮＳ）　削　除（ＤＥＬ）

ラインアップ・ダウン（↓，↑）

メモリー保護……バッテリーバックアップ

（プログラム、データ、リザーブ内容など電源オフ時内容保護）

表　　示……ミニグラフィック表示　：　７×156ドット

文字表示　：　26桁

使　用　温　度……０℃～40℃

電　　源……６Ｖ ⎓（ＤＣ）　：　単３タイプ乾電地４本

電池使用時間……単３タイプ乾電池（ＳＵＭ-3）　約50時間

（使用温度20℃で5555555555555.を連続的に表示させる場合。電池の種類や使用方法などにより多少の変動があります。）

● １日当り１時間使用した場合は約40日使用できます。ただし１時間のうち、演算またはプログラム実行を10分間、表示状態を50分間おこなった場合です。

消　費　電　力……0.13Ｗ

外　形　寸　法……幅195㎜×奥行86㎜×高さ25.5㎜

重　　量……375ｇ（電池含む）

付　　属　品……ソフトケース、テンプレート２枚、単３乾電池４本、取扱説明書、ネームラベル、シャープサービス・お客様ご相談窓口一覧表

▶ポケットコンピュータ友の会御案内◀

入会手続

1．皆様の作成されました興味あるオリジナルプログラム一編以上を巻末の所定フォーム（のコピー）を用いて、本取扱説明書中のプログラム事例集に従がい記載いただき、下記送付先へお送り下さい。

（注）御送付頂いたプログラムは返却致しませんので御了承ください。また御送付頂いたプログラムの製本化やカセットテープ化等に伴なう編集及び出版販売等に関する一切の権利は弊社（シャープ㈱）に無償で譲度頂きます。プログラムは、オリジナルなもので、弊社（シャープ㈱）以外への２重投稿はお断わり致します。

会員特典

1．会員証を発行致します。（有効期間１年）

1．御送付頂いたプログラムは、弊社（シャープ㈱）で選考の上、製本化し、出版会社を通じて出版・販売致します。尚、採用プログラム分につきましては記念品を贈呈させて頂きます。

1．上記の本を作る際の該当期間中に、オリジナルプログラムをお送り頂いた会員の皆様には、この本を無償で贈呈致します。

送付先　〒639-11　奈良県大和郡山市美濃庄町492

シャープ株式会社　情報システム事業本部

コンピュータ事業本部　ポケットコンピュータグループ

プログラム事例集

このページまで各種機能の説明を読み進まれ、多くのプログラム命令の知識を習得されたと思います。しかしながら、ＢＡＳＩＣ言語で自由自在にプログラム作成ができるようになるには、この取扱説明書に加え、実際的なプログラムを自分で作成してみることが必要です。車の運転やテニスなどと同じで、自分自身で実際にハンドルをにぎり、ラケットを振って練習することにより、初めて上達するように、プログラムも上手、下手を気にせずに、数多く作成することにより初めて上達します。他人の作成したプログラムを参考にすることも大切です。

以降のページにＢＡＳＩＣの各命令をいろいろ使用したプログラムをいくつか載せていますので参考にしてください。

（なお、当社はこのプログラムの使用に際して生ずる金銭上の損害および逸失利益については、一切の責任を負いませんのであらかじめご了承ください。）

プログラムライブラリーのご案内

- ＰＣ-1500用　ソフトウェア・ライブラリ　2　（既刊）
- ＰＣ-1500用　ソフトウェア・ライブラリ　3　（既刊）
- ＰＣ-1500用　ソフトウェア・ライブラリ　6　（既刊）
- ＰＣ-1501用　ソフトウェア・ライブラリ　7　（既刊）
- ＰＣ-1500用　プログラムライブラリ　Series−2　（既刊）

事務分野や数学統計・技術さらにはゲーム分野などで実用性と楽しさ、また、はん用性に優れたプログラムを編集し、出版社を通じて出版販売致しております。ＰＣ-1500用のライブラリもＰＣ-1501用としてお使いいただけます。くわしくはお近くの販売店または有名書店にお問い合わせください。

目　次

（プログラム名）　（ページ）

●方程式の根…… 122

●相関係数，1次回帰，プロット…… 126

●借入限度額、ローン返済回数の計算…… 133

●並びかえ…… 136

●度数分布図…… 141

●バイオリズム…… 146

●タイプ練習…… 151

●ストップウオッチ,タイマー,目覚し時計…… 154

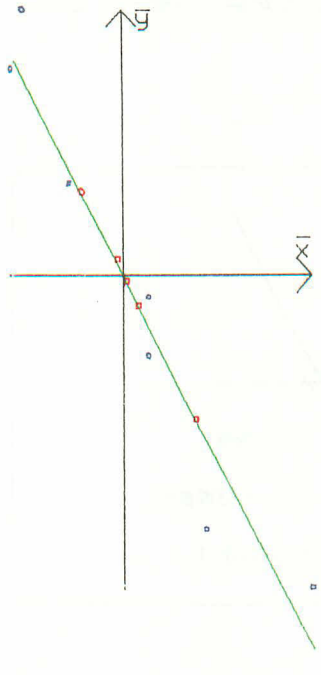
●コンピュータ・フラワー …… 160

●コンピュータ・グラフィック …… 163

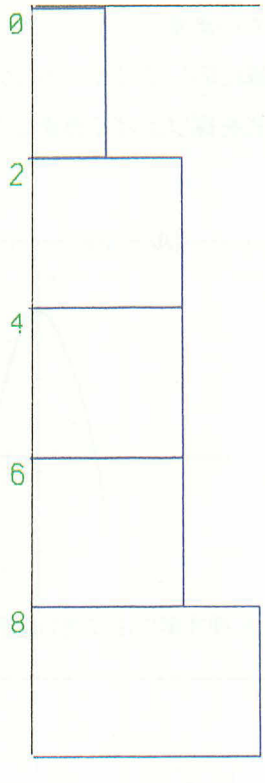
このプログラム事例集に集録されているプログラムはLPRINT、LINE、COLORなど、プリンタに関する命令が使用されているものがあります。これらのプリンタに関する命令は、別売のカラーグラフィックプリンタ〈CE-150〉が持っている命令ですので、PC-1501のみで、これらの命令を入力した場合は、エラーあるいはプログラムの誤動作の原因になります。(プリンタに関する命令については84ページを参照ください。)

プリント出力例

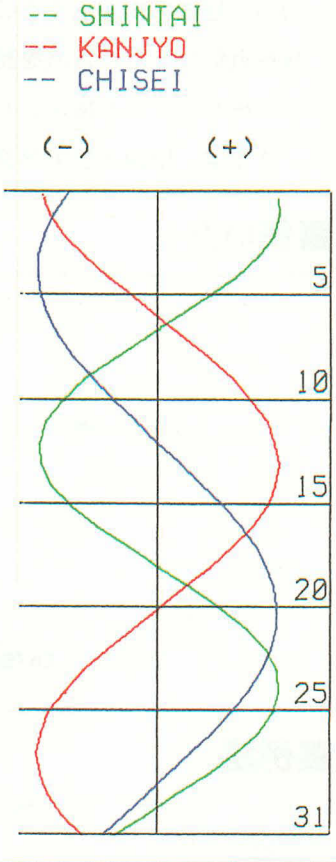
★相関係数、1 次回帰、プロット（P.127 参照）



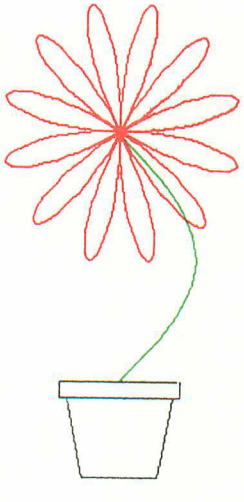
★度数分布図（P.142 参照）



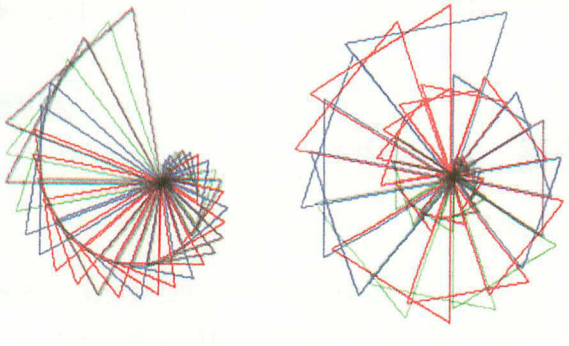
★バイオリズム（P.147参照）
（身体、感情、知性）



★コンピュータ・フラワー
（P.160 参照）



★コンピュータ・グラフィック（P.164 参照）



PROGRAM TITLE

方程式の根

1

概要（数学分野）

一般の方程式の根を求めるのはなかなか面倒なものですが、ここではニュートン法を用いて根を近似して行く方法を紹介しています。

一ツの根が求まると指定した区間幅に応じてニュートン法を用いる始点が自動的に変わっていきます。例題として３次方程式を採り上げてみました。

■使い方

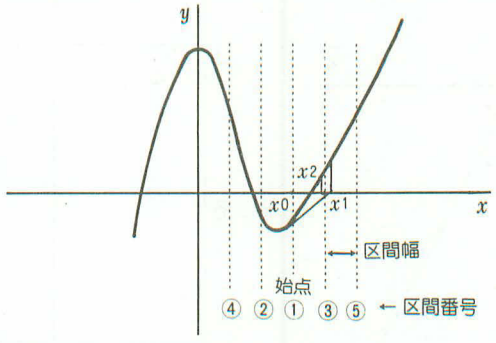
《ルール》

入力

始点

微小区間

区間幅



出力

根の値（ENTERキーを押す事により次の区間での根を求めます。）

■例題

$x^3-2x^2-x+2=0$（根は−１，１，２）

始点＝0

微小区間＝10^{-4}

区間幅＝0.5

として計算します。

関数は、サブルーチンとして、500ライン以後に書き込んで下さい。

書き込み方（例題の場合）　１．PROモードにする。（MODEキーを押して）

２．500B＝((X−2)＊X−1)＊X＋2 ENTER

510RETURN ENTER　これで書込み終了。

■内容　（計算内容など）

$$Xn+1=Xn-\frac{f(Xn)}{f'(Xn)}$$

XnとXn+1の差の絶対値が10^{-8}以下になればXnを根とみなし表示します。

ここでは１次微分のf'(X)を次の様に定義します。

$$f'(X)=\frac{f(X+h)-f(X)}{h}$$　（hは微小区間）

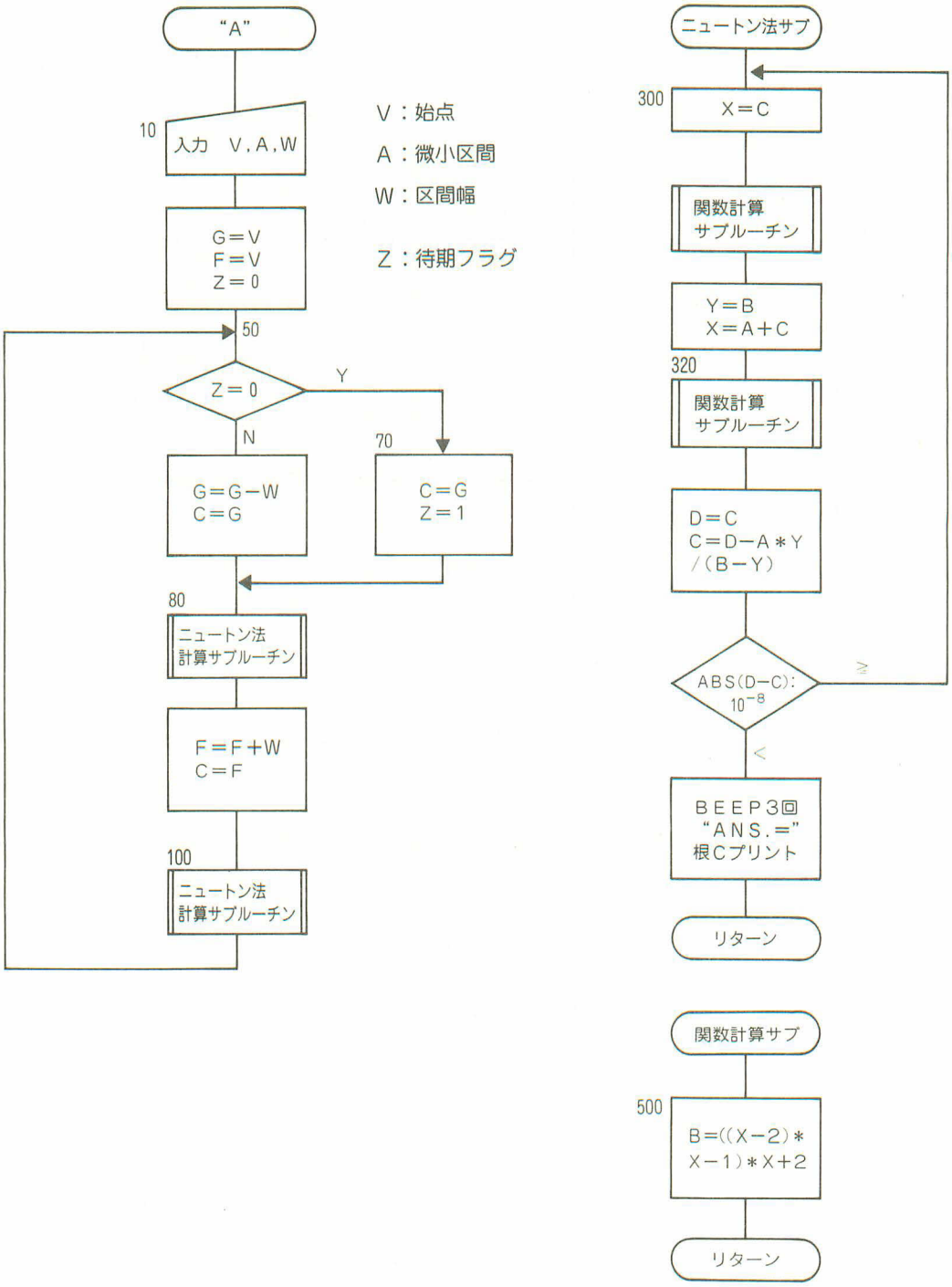
10^{-8}を変更する時は、340ラインの１E−８を変更して下さい。

PROGRAM TITLE

方程式の根

2

■フローチャート



PROGRAM TITLE

方程式の根

3

■キー操作手順

ステップ	入　力	表　示	備　考
1	DEF A	SHITEN=	
2	0 ENTER	BISHO=	
3	0.0001 ENTER	KUKAN=	
4	0.5 ENTER	ANS.=2	
5	ENTER	ANS.=1	ENTER を繰り返し複数の根を求めます。
6	ENTER	ANS.=−1	
	ENTER	ANS.=1	
	ENTER	ANS.=−1	
	ENTER	ANS.=−1	
	ENTER	ANS.=−1	
	ENTER	ANS.=2	
		ANS.=−1	

PROGRAM TITLE

方程式の根

4

■プログラムリスト ※文字の大きさは18字／行

```
 10:"A":INPUT "SHI
    TEN=";U
 20:INPUT "BISHO="
    ;A
 30:INPUT "KUKAN="
    ;W
 40:G=U:F=U:Z=0
 50:IF Z=0GOTO 70
 60:G=G-W:C=G:GOTO
    80
 70:C=G:Z=1
 80:GOSUB 300
 90:F=F+W:C=F
100:GOSUB 300
110:GOTO 50
120:END
300:X=C:GOSUB 500
310:Y=B:X=A+C
320:GOSUB 500
330:D=C:C=D-A*Y/(B
    -Y)
340:IF ABS (D-C)>=
    1E-8GOTO 300
350:BEEP 3:PRINT "
    ANS.=";C
360:RETURN
500:B=((X-2)*X-1)*
    X+2
510:RETURN
```

メモリー内容	
A	微小区間(入力値)＝h
B	f(X)
C	X_0
D	f(X＋h)
E	
F	√
G	√
H	
I	
J	
K	
L	
M	
N	
O	
P	
Q	
R	
S	
T	
U	
V	始点(入力値)
W	区間幅(入力値)
X	X
Y	f(X)
Z	初期フラグ

PROGRAM TITLE

相関係数,1次回帰,プロット

1

カラーグラフィック
プリンター(CE-150)必要

概要(統計分野)

データは分析され、推定されるためにある。

このプログラムは2ツの関連したデータ(X_1, Y_1)…(X_n, Y_n)の間の共分散と相関係数を求め、更に1次回帰式も求めます。その後与えられたデータに対してY＝AX＋Bへのあてはめ(推定)を行ない、その結果もプリンターにグラフ化します。

■使い方

1．データの入力(X_i，Y_i)(但し本体標準容量で$i \leq 256$)

2．共分散、相関係数、回帰係数、平均値を求めプリンターに出力。

3、$\overline{X}$，$\overline{Y}$をX軸、Y軸とするグラフに表現し、入力データ及び推定値は色を変えてプロットします。

4．推定値XからYを求め、X，Yの値をプリンターに出力します。

■例題

X	6.9	7.6	7.6	9.0	8.1	6.5	6.4	6.9
Y	12	10	9	5	6	15	14	12

共分散＝－3.060714286

相関係数＝－9.693968513E－01

回帰係数a＝－3.942042318

b＝39.4475621

平均値$\overline{X}$＝7.375

$\overline{Y}$＝10.375

推定値X＝7，　Y＝11.85326587

X＝8，　Y＝7.911223556

X＝7.5，　Y＝9.882244715

X＝7.3，　Y＝10.67065318

X＝7.4，　Y＝10.27644895

. . . . . .

PROGRAM TITLE

相関係数, 1次回帰, プロット

2

■内容（計算内容など）

$Sxx=\Sigma xi^2-n\bar{x}^2$

$Sxy=\Sigma xiyi-n\bar{x}\bar{y}$

$Syy=\Sigma yi^2-n\bar{y}^2$

$C \quad =Sxy/(n-1)$ ……共分散

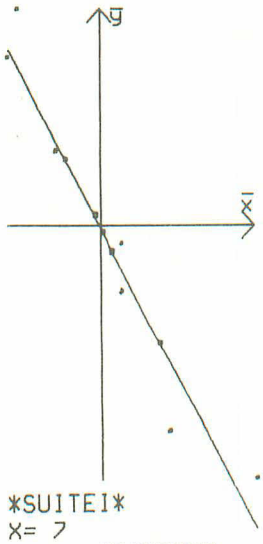
$r \quad =Sxy/\sqrt{SxxSyy}$ ……相関係数

$a \quad =Sxy/Sxx$
$b \quad =\bar{y}-a\bar{x}$
} 回帰係数（$y=ax+b$）

■プリント出力　※文字の大きさは18字／行

（実際はカラープリント出力です。P.121参照）

```
KYOBUNSAN=
       -3.060714286
SOUKAN=
  -9.693968513E-01
KAIKI-KEISUU
A=-3.942042318
B= 39.4475621
*HEIKIN*
X= 7.375
Y= 10.375
```



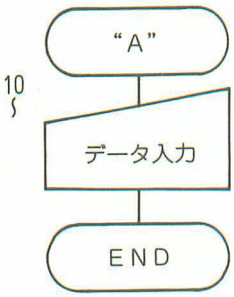

```
*SUITEI*
X= 7
Y= 11.85326587
X= 8
Y= 7.911223556
X= 7.5
Y= 9.882244715
X= 7.3
Y= 10.67065318
X= 7.4
Y= 10.27644895
```

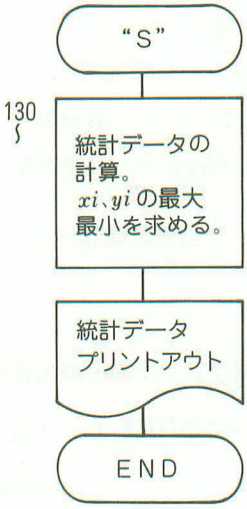
PROGRAM TITLE

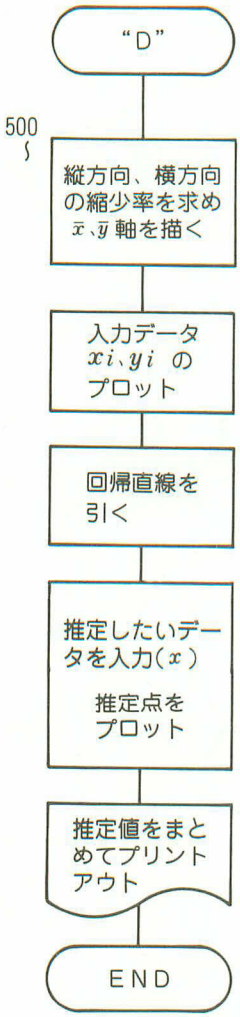
相関係数, 1次回帰, プロット

3

■概略フローチャート





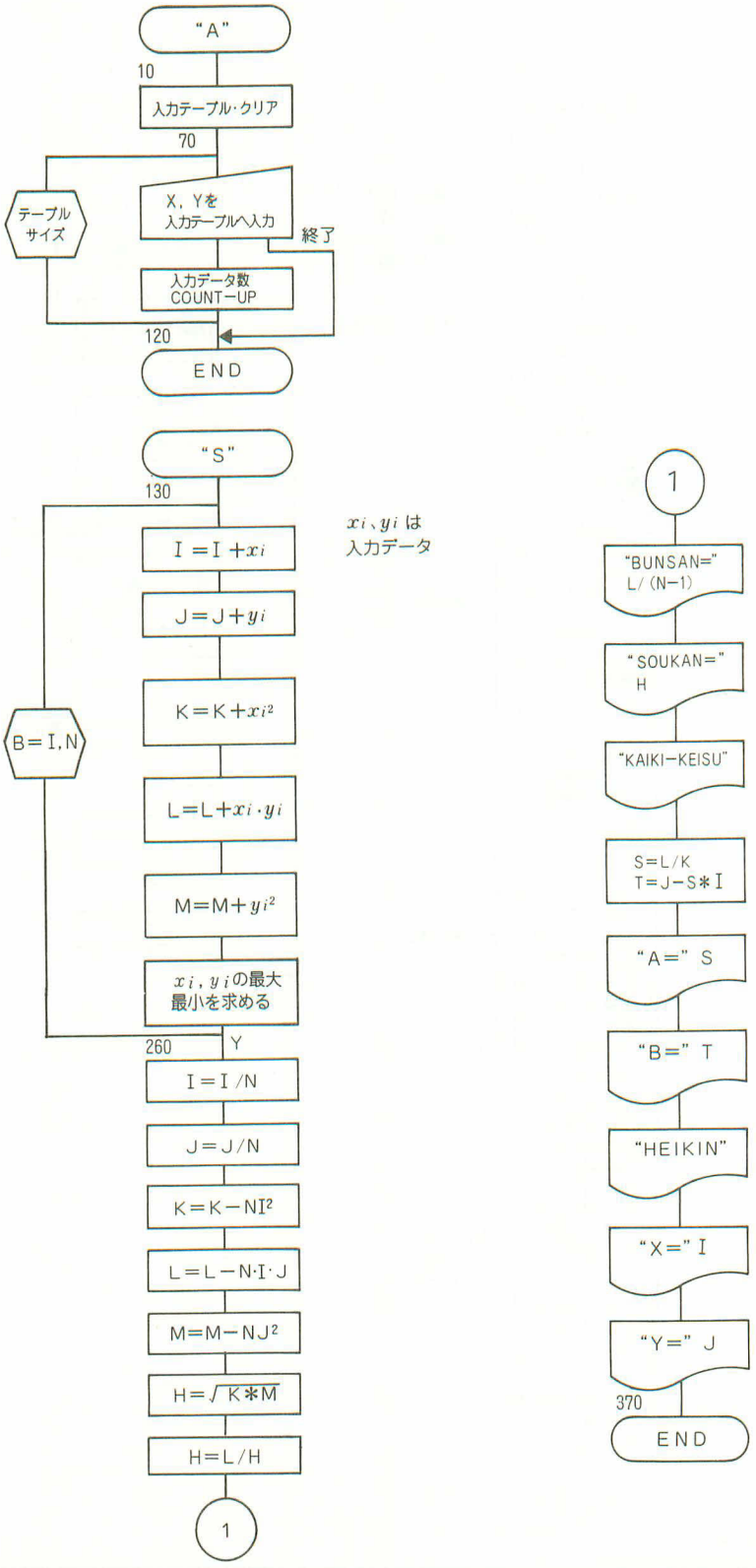


PROGRAM TITLE

相関係数, 1次回帰, プロット

3

■フローチャート

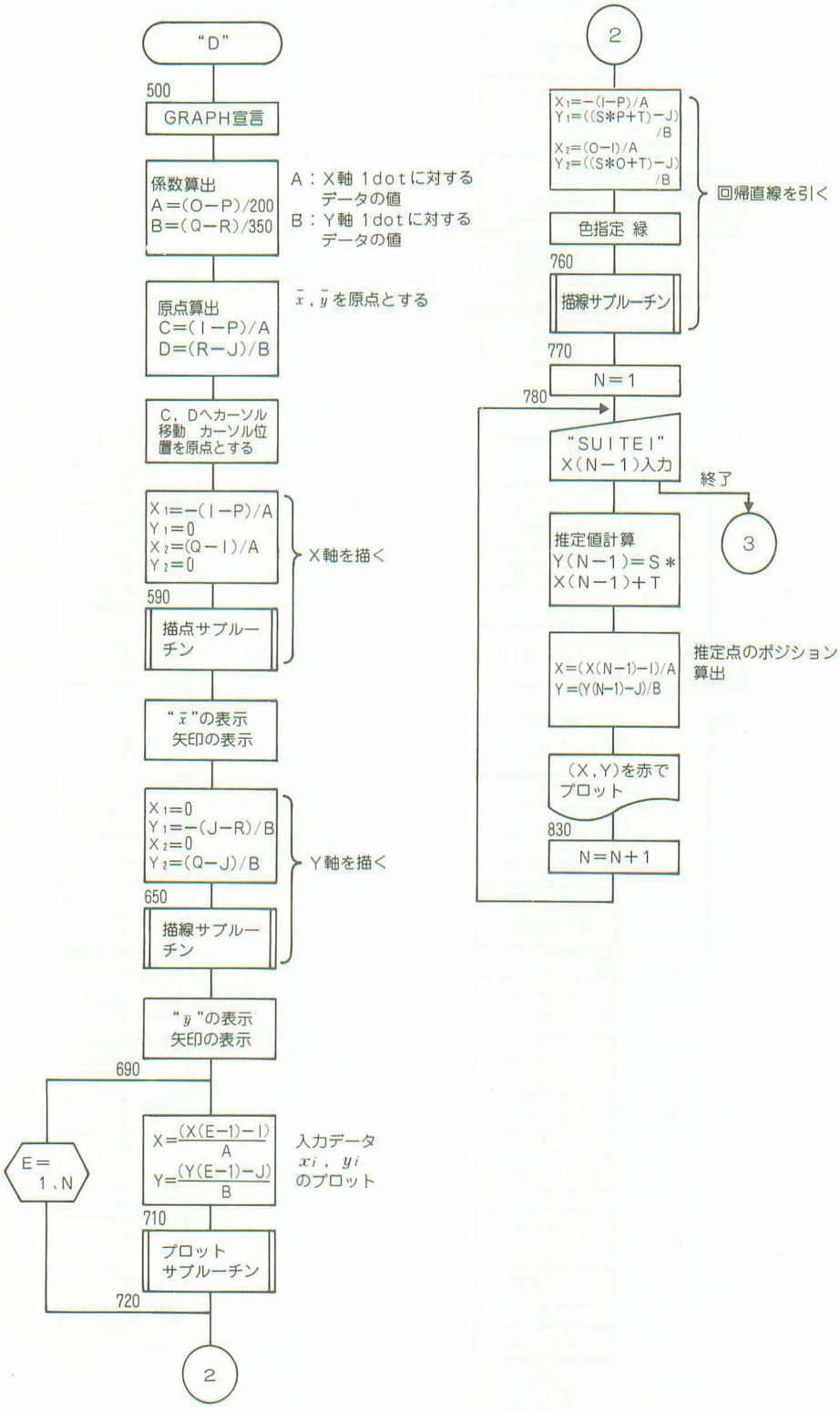


PROGRAM TITLE

相関係数, 1次回帰, プロット

3

■フローチャート

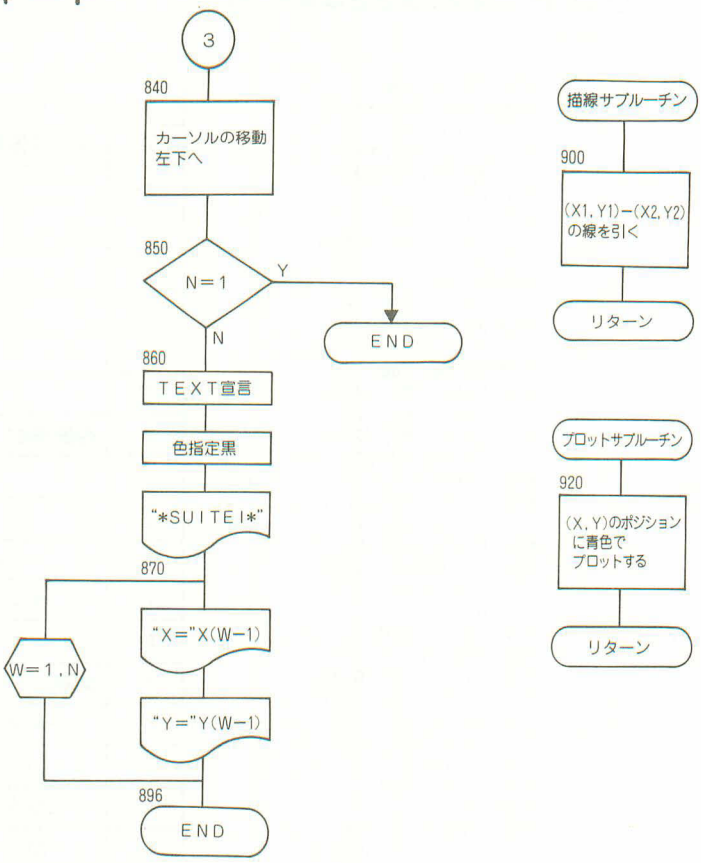


PROGRAM TITLE

相関係数,1次回帰,プロット

3

■フローチャート



■キー操作手順

ステップ	キー入力	表示	備考
1	DEF A	X=	
2	6.9 ENTER	Y=	(1)の表示に戻り(2)で ENTER キーを押すか、又は10組分のデーターを入力する迄繰返します。
3	12 ENTER	X=	
⋮	⋮	⋮	
18	ENTER	>	
19	DEF S	>	>が表示されれば終了。この間にプリンターに分散等を出力します。
20	DEF D	SUITEI=	表示が出る迄にプリンターにグラフを出力します。
21	7 ENTER	SUITEI=	(20)の表示に戻り10データ分入力するか、又は ENTER のみ入力する迄繰返します。
⋮	⋮	⋮	

PROGRAM TITLE

相関係数, 1次回帰, プロット

4

■プログラムリスト ※文字の大きさは18字／行

```
 10:"A":CLEAR
 20:DIM X(255),Y(2
    55)
 30:FOR B=1TO 256
 40:X(B-1)=0:Y(B-1
    )=0
 50:NEXT B
 60:N=0
 70:FOR B=1TO 256
 80:INPUT "X=";X(B
    -1):GOTO 95
 90:GOTO 120
 95:INPUT "Y=";Y(B
    -1)
100:N=N+1
110:NEXT B
120:END
130:"S":I=0:J=0:K=
    0:L=0:M=0
140:P=10^(98):Q=-P
    :R=P:Q=0
150:FOR B=1TO N
155:Z=B-1
160:I=I+X(Z)
170:J=J+Y(Z)
180:K=K+X(Z)*X(Z)
190:L=L+X(Z)*Y(Z)
200:M=M+Y(Z)*Y(Z)
210:IF P>X(Z)LET P
    =X(Z)
220:IF Q<X(Z)LET Q
    =X(Z)
230:IF R>Y(Z)LET R
    =Y(Z)
240:IF Q<Y(Z)LET Q
    =Y(Z)
250:NEXT B
260:I=I/N:J=J/N
270:K=K-N*I*I
280:L=L-N*I*J
290:M=M-N*J*J
305:H=SQR(K*M)
307:H=L/H
310:COLOR 0:LPRINT
    "KYOBUNSAN=",L
    /(N-1)
320:LPRINT "SOUKAN
    =",H
330:LPRINT "KAIKI-
    KEISUU"
340:S=L/K:T=J-S*I
350:LPRINT "A=";S
360:LPRINT "B=";T
362:LPRINT "*HEIKI
    N*"
364:LPRINT "X=";I
366:LPRINT "Y=";J
370:END
500:"D":GRAPH
510:A=(Q-P)/200
520:B=(Q-R)/350
530:C=(I-P)/A
540:D=(R-J)/B
550:GLCURSOR (C,D)
560:SORGN
570:X1=-(I-P)/A:Y1
    =0
580:X2=(Q-I)/A:Y2=
    0
590:GOSUB 900
600:LINE (X2-10,Y2
    -10)-(X2,Y2)
605:LINE (X2,Y2)-(
    X2-10,Y2+10)
610:LPRINT "x"
620:LINE (X2-10,Y2
    +23)-(X2,Y2+23
    )
630:X1=0:Y1=-(J-R)
    /B
640:X2=0:Y2=(Q-J)/
    B
650:GOSUB 900
660:LINE (X2-10,Y2
    -10)-(X2,Y2)
665:LINE (X2,Y2)-(
    X2+10,Y2-10)
670:LPRINT "y"
680:LINE (X2+10,Y2
    )-(X2+20,Y2)
690:FOR E=1TO N
700:X=(X(E-1)-I)/A
    :Y=(Y(E-1)-J)/
    B
710:GOSUB 920
720:NEXT E
730:X1=-(I-P)/A:Y1
    =((S*P+T)-J)/B
740:X2=(Q-I)/A:Y2=
    ((S*Q+T)-J)/B
750:COLOR 2
760:GOSUB 900
770:N=1
780:INPUT "SUITEI=
    ";X(N-1):GOTO
    800
790:GOTO 840
800:Y(N-1)=S*X(N-1
    )+T
810:X=(X(N-1)-I)/A
    :Y=(Y(N-1)-J)/
    B
820:LINE (X-1,Y-1)
    -(X+2,Y+2),0,3
    ,B
830:N=N+1:GOTO 780
840:GLCURSOR (-(I-
    P)/A,-(J-R)/B-
    20)
850:IF N=1END
852:TEXT
860:COLOR 0:LPRINT
    "*SUITEI*"
870:FOR W=1TO N-1
880:LPRINT "X=";X(
    W-1)
890:LPRINT "Y=";Y(
    W-1)
895:NEXT W
896:END
900:LINE (X1,Y1)-(
    X2,Y2)
910:RETURN
920:LINE (X,Y)-(X+
    2,Y+2),0,1,B
930:RETURN
```

メモリー内容	
A	グラフ係数(1dot移動)X
B	〃 Y
C	√
D	√
E	√
F	
G	
H	相関係数
I	$\overline{X}$
J	$\overline{Y}$
K	$S_{xx}=\Sigma X_i^2-n\overline{X}^2$
L	$S_{xY}=\Sigma X_i \cdot Y_i-n\overline{X}\,\overline{Y}$
M	$S_{YY}=\Sigma Y_i^2-n\overline{Y}^2$
N	個数(データ)n
O	X−MAX
P	X−MIN
Q	Y−MAX
R	Y−MIN
S	回帰係数 a
T	〃 b
U	
V	
W	√
X	√
Y	√
Z	√
X(9)	入力、推定データ格納
Y(9)	入力、推定データ格納
X_1	描線SUB用スタートX
Y_1	X Y
X_2	〃 エンドX
Y_2	〃 Y

PROGRAM TITLE

借入限度額,ローン返済回数の計算

1

カラーグラフィック
プリンター(CE-150)必要

概要(事務分野)

無理な借金は禁物。

適切な借り入れのための借入限度額の計算と、自分の返済能力から見た返済回数の計算が行なえます。地道な返済プランを……。

■使い方

"A" 返済能力(返済額と返済回数)から、借入限度額を計算します。(1円未満切り捨て)

"B" 借入金額と返済能力から返済回数を計算します。

■例題

"A" 返済期間8年、返済能力は毎月150万円、年利率12%なら、借入限度額はいくらになるか計算します。

毎年の返済額=1500000*12
返済年数　=8
利　率　=12%
} 入力

"B" 300万円を借り、年利が12%で返済能力が月10万円であるなら何ヶ月で返済できるか計算します。

借入金額　=3000000
毎月の返済額=100000
利　率　=12/12%
} 入力

■内容(計算内容など)

	借入限度額計算	ローン返済回数の計算
入　力	毎回の返済額(a円) ローン返済回数(n回) 利　率(r%)	借入金額(A円) 毎回の返済額(a円) 利　率(r%)
出　力	借入限度額	ローン返済回数

$$借入限度額=\frac{a(R^n-1)}{(R-1)R^n}$$

(ただし　$R=1+\frac{r}{100}$　)

$$返済回数=\frac{\log a-\log(a-A(R-1))}{\log R}$$

PROGRAM TITLE

借入限度額,ローン返済回数の計算

2

■プリント出力 ※文字の大きさは18字／行

(A)

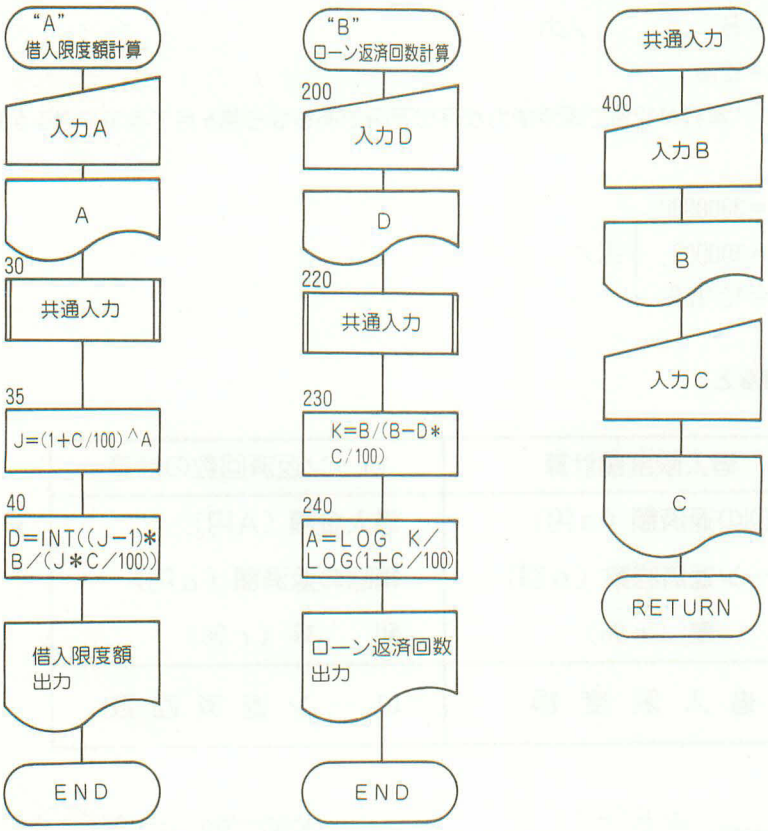
```
HENSAIKAISU=   8.00
HENSAIGAKU=
        18,000,000
RIRITU(%)=   12.000
GENDOGAKU=
        89,417,515
```

(B)

```
GENDOGAKU=
         3,000,000
HENSAIGAKU=
           100,000
RIRITU(%)=    1.000
HENSAIKAISU= 35.84
```

■フローチャート

A：ローン返済回数
B：毎回の返済額
C：利率(%)
D：借入限度額



PROGRAM TITLE

借入限度額,ローン返済回数の計算

3

■キー操作手順

ステップ	キー入力	表示	備考
1	DEF A	HENSAIKAISU?	
2	8 ENTER	HENSAIGAKU?	
3	1500000＊12 ENTER	RIRITSU(%)?	
4	12 ENTER	>	
1	DEF B	GENDOGAKU?	
2	3000000 ENTER	HENSAIGAKU?	
3	100000 ENTER	RIRITSU(%)?	
4	12／12 ENTER	>	

■プログラムリスト ※文字の大きさは18字／行

```
 10:"A"CLEAR :LF 2
 20:INPUT "HENSAIK
    AISU? ";A
 25:LPRINT "HENSAI
    KAISU=";USING
    "###.##";A
 30:GOSUB 400
 35:J=(1+C/100)^A
 40:D=INT ((J-1)*B
    /(J*C/100))
 50:LPRINT "GENDOG
    AKU="
 55:LPRINT USING "
    ############,#
    ##";D
 60:LF 3:END
200:"B"CLEAR :LF 2
210:INPUT "GENDOGA
    KU? ";D
215:LPRINT "GENDOG
    AKU=":LPRINT
    USING "#######
    ##,###";D
220:GOSUB 400
230:K=B/(B-D*C/100
    )
240:A=LOG K/LOG (1
    +C/100)
250:LPRINT "HENSAI
    KAISU=";USING
    "###.##";A
260:LF 3:END
400:INPUT "HENSAIG
    AKU? ";B
405:LPRINT "HENSAI
    GAKU=":LPRINT
    USING "#######
    #####,###";B
410:INPUT "RIRITU(
    %)? ";C
415:LPRINT "RIRITU
    (%)=";USING "#
    ###.###";C
430:RETURN
440:END
```

メモリー内容	
A	ローン返済回数
B	毎回の返済額
C	利率（%）
D	借入限度額
E	
F	
G	
H	
I	
J	計算WORK
K	計算 〃
L	
M	
N	
O	
P	
Q	
R	
S	
T	
U	
V	
W	
X	
Y	
Z	

PROGRAM TITLE

並びかえ

1

カラーグラフィック
プリンター(CE-150)必要

概要

高速並びかえ！

業績や順位、名簿などは、手計算で並びかえると非常に時間がかかります。このような煩わしさは、このプログラムで解消し、スピーディーな事務処理をしましょう。

■使い方

まず並びかえたいデータが、文字か数値かの選択、そしてデータ数を入力します。次に、小さい順に並びかえるか、又は大きい順に並びかえるかを選択し(文字の場合はA、B、C、…が小さい順で、Z、Y、X、…が大きい順です。)データを入力しますと、並びかえられたデータがプリントされます。

■例題

数　値
データ数　7
小さい順
データ
- 122
- 123
- 1
- 123456
- 1234
- 11
- 12345

文　字
データ数　5
小さい順
データ
- CCR
- ABCDE
- RGHI
- KTV
- CCK

■内容

入力●文字か数値かの選択
- ●データ数(標準容量で数値の場合256データまで、文字の場合256データまで)
- ●小さい順か大きい順の選択
  (文字の場合はアスキーコードの大小に従います)
- ●文字か数値のデータ

出力●並びかえられたデータ

注意：文字データの文字列は16文字までです。

PROGRAM TITLE

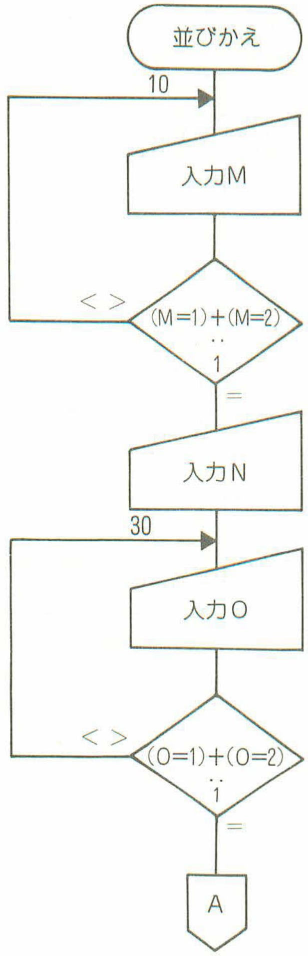
並びかえ 2

■プリント出力

```
      1
     11
    122
    123
   1234
  12345
 123456

ABCDE
CCK
CCR
KTV
RGHI
```

■フローチャート



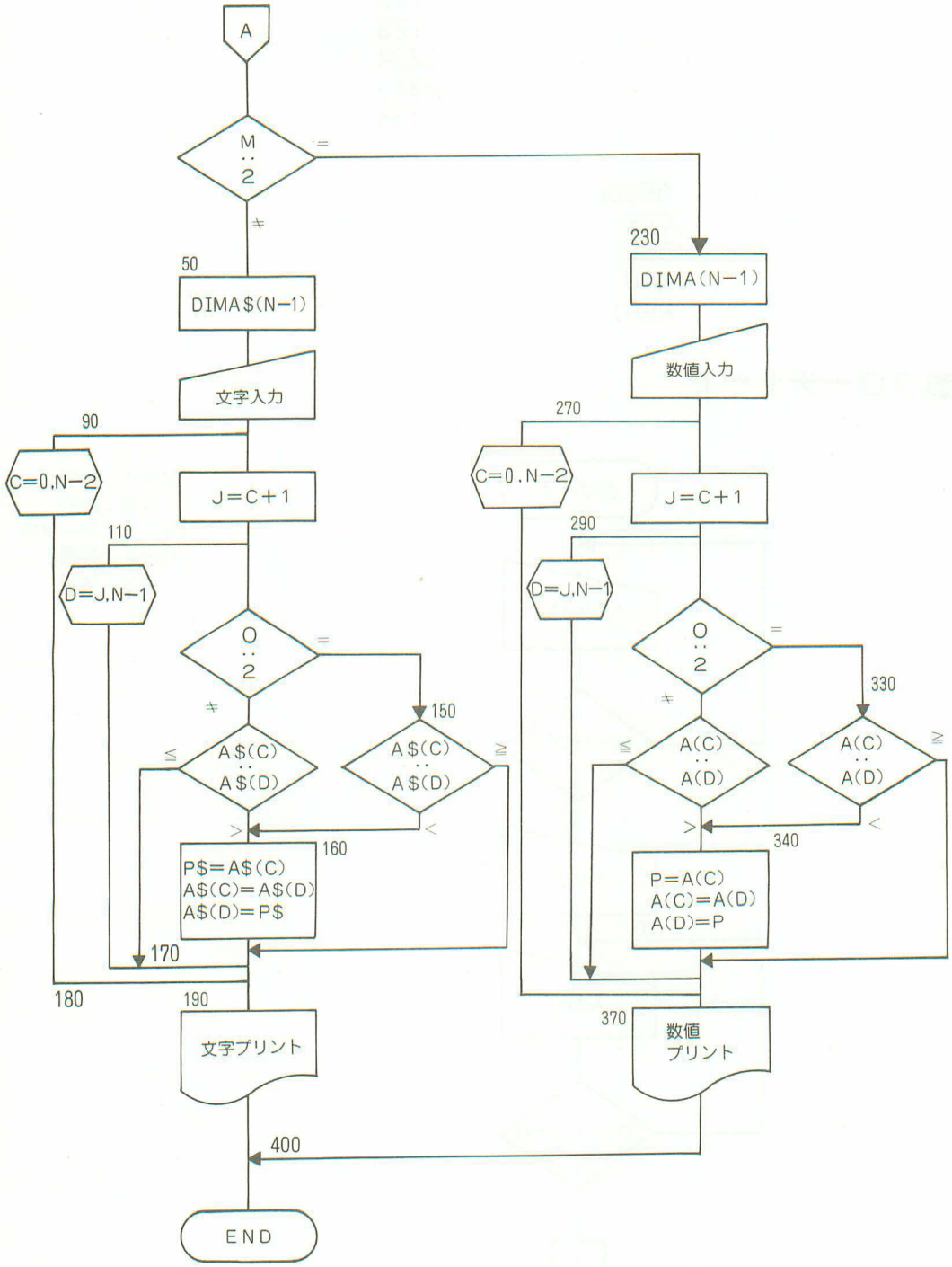
M=1：文字
　=2：数字
N　：データ数
O=1：昇順
　=2：降順

PROGRAM TITLE

並びかえ

3

■フローチャート



PROGRAM TITLE

並びかえ

4

■キー操作入力

ステップ	入力キー	表示	備考
1	DEF A	MOJI=1 SUUJI=2?	
2	2 ENTER	DATA SUU?	数値の並びかえを選択
3	7 ENTER	CHIISAI=1 OOKII=2?	
4	1 ENTER	X=	
5	122 ENTER	X=	
6	123 ENTER	X=	
7	1 ENTER	X=	
8	123456 ENTER	X=	
9	1234 ENTER	X=	
10	11 ENTER	X=	
11	12345 ENTER	>	プリント出力
1	DEF A	MOJI=1 SUUJI=2?	
2	1 ENTER	DATA SUU?	文字の並びかえを選択
3	5 ENTER	CHIISAI=1 OOKII=2?	
4	1 ENTER	X$=	
5	CCR ENTER	X$=	
6	ABCDE ENTER	X$=	
7	RGHI ENTER	X$=	
8	KTV ENTER	X$=	
9	CCK ENTER	>	プリント出力

PROGRAM
TITLE

並びかえ

5

■プログラムリスト ※文字の大きさは18字／行

```
 10:"A":CLEAR :
    INPUT "MOJI=1
    SUUJI=2 ?";M
 15:IF (M=1)+(M=2)
    <>1GOTO 10
 20:INPUT "DATA SU
    ?";N
 30:INPUT "CHIISAI
    =1 OOKII=2 ?";
    O
 35:IF (O=1)+(O=2)
    <>1GOTO 30
 40:IF M=2GOTO 230
 50:DIM A$(N-1)
 60:FOR B=0TO N-1
 70:INPUT "X$=";A$
    (B)
 80:NEXT B
 90:FOR C=0TO N-2
100:J=C+1
110:FOR D=JTO N-1
120:IF O=2GOTO 150
130:IF A$(C)>A$(D)
    GOTO 160
140:GOTO 170
150:IF A$(C)<A$(D)
    GOTO 160
155:GOTO 170
160:P$=A$(C):A$(C)
    =A$(D):A$(D)=P
    $
170:NEXT D
180:NEXT C
190:FOR E=0TO N-1
200:LPRINT A$(E)
210:NEXT E
220:GOTO 400
230:DIM A(N-1)
240:FOR B=0TO N-1
250:INPUT "X=";A(B
    )
260:NEXT B
270:FOR C=0TO N-2
280:J=C+1
290:FOR D=JTO N-1
300:IF O=2GOTO 330
310:IF A(C)<=A(D)
    GOTO 350
320:GOTO 340
330:IF A(C)>=A(D)
    GOTO 350
340:P=A(C):A(C)=A(
    D):A(D)=P
350:NEXT D
360:NEXT C
370:FOR E=0TO N-1
380:LPRINT A(E)
390:NEXT E
400:END
```

メモリー内容					
A		A$		A (N)	数値データ
B	ループ処理カウンター	B$		A$(N)	文字データ
C	〃 〃	C$			
D	〃 〃	D$			
E	〃 〃	E$			
F		F$			
G		G$			
H		H$			
I		I$			
J	√	J$			
K		K$			
L		L$			
M	文字、数字の選択コード	M$			
N	データ数	N$			
O	昇順、降順選択コード	O$			
P	並びかえ数字の置換用	P$	並びかえ文字の置換用		
Q		Q$			
R		R$			
S		S$			
T		T$			
U		U$			
V		V$			
W		W$			
X		X$			
Y		Y$			
Z		Z$			

PROGRAM TITLE

度数分布図

1

カラーグラフィックプリンター(CE-150)及びカセットレコーダー(CE-152)必要

概要（事務分野）

得られたデータを、いくつかのランクに分け、それぞれどれぐらいの度数があるか知ることは、全体像を把握するためにしばしば必要になることです。度数分布図を描きますので、視覚的にとらえることもできます。

■使い方

1．パラメータの入力(データ数、級の初期値、級の幅、級の数)

2．データの入力方法設定〔キー入力又はカセット入力〕

キー入力　　：データをキーから入力し、その後カセット･テープへデータ出力

カセット入力：データをカセットテープから入力します。

3．分散及び標準偏差を計算し、プリンターへ出力します。

4．度数分布図をプリンターへ出力します。

注）データは最大256までとします。

■例題

データ数＝10、級の初期値＝０、級の幅＝２、級の数＝５

5	2	7	9	8	1	3	4	6	8

分散　　：6.81

標準偏差：2.60959767

■内容（計算内容など）

$$V=\frac{1}{n}\sum_{i=1}^{n}(xi-\bar{x})^2 \quad \text{（分散）}$$

$$S=\sqrt{V} \quad \text{（標準偏差）}$$

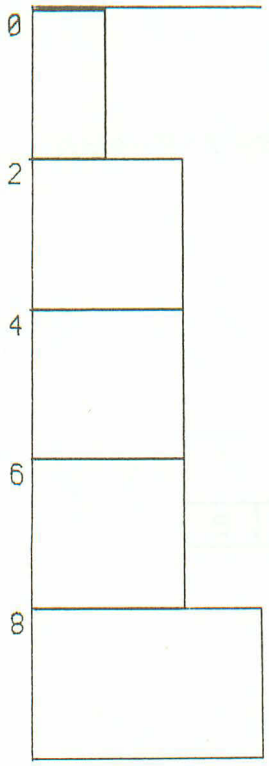
PROGRAM TITLE 度数分布図 2

■プリント出力

※文字の大きさは18字／行

（実際はカラープリント出力です。P.121参照）

```
BUNSAN= 6.81
HYOJUN HENSA=
          2.60959767
```



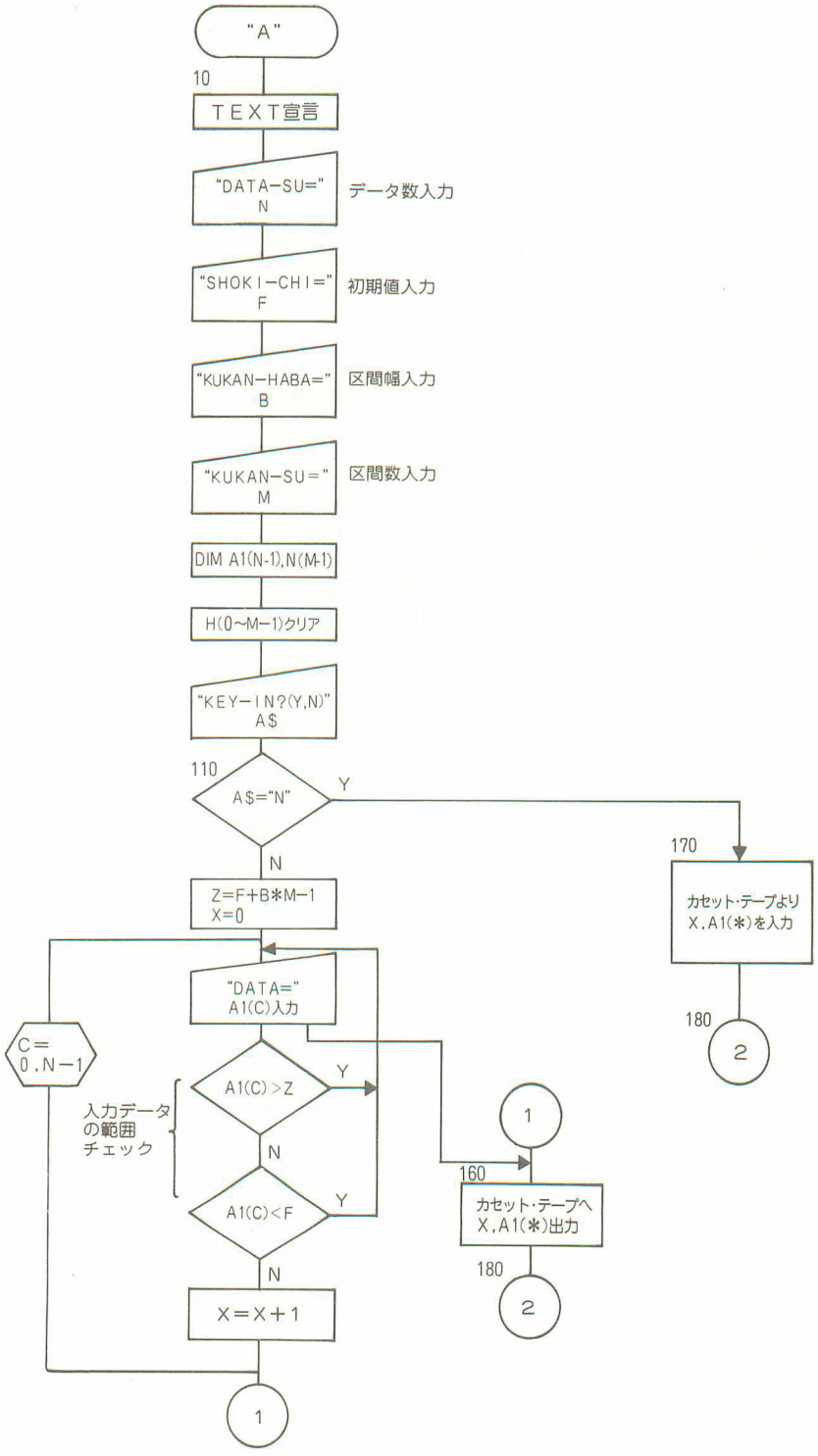
■プログラムリスト

※文字の大きさは18字／行

```
10:"A":CLEAR :
   TEXT :USING
20:INPUT "DATA-SU
   =";N
30:INPUT "SHOKI-C
   HI=";F
40:INPUT "KUKAN-H
   ABA=";B
50:INPUT "KUKAN-S
   U=";M
60:DIM A1(N-1),H(
   M-1)
70:FOR C=0TO M-1
80:H(C)=0
90:NEXT C
100:INPUT "KEY-IN?
    (Y,N)";A$
110:IF A$="N"GOTO
    170
115:Z=F+B*M-1:X=0
120:FOR C=0TO N-1
130:INPUT "DATA=";
    A1(C):GOTO 150
140:GOTO 160
150:IF A1(C)>ZGOTO
    130
152:IF A1(C)<FGOTO
    130
153:X=X+1
155:NEXT C
160:PRINT #X,A1(*)
165:GOTO 180
170:INPUT #X,A1(*)
180:S=0:N=X
190:FOR C=0TO N-1
200:I=INT ((A1(C)-
    F)/B)
210:H(I)=H(I)+1
220:S=S+A1(C)
230:NEXT C
240:V=S/N:T=0
250:FOR C=0TO N-1
260:T=T+(A1(C)-V)^
    2
270:NEXT C
280:T=T/N:S=√T
290:COLOR 0:LPRINT
    "BUNSAN=";T
300:LPRINT "HYOJUN
     HENSA=",S
310:N=-10^(98)
320:FOR C=0TO M-1
330:IF H(C)>NLET N
    =H(C)
340:NEXT C
350:GRAPH
360:GLCURSOR (50,0
    ):SORGN
370:COLOR 0
380:LINE (0,0)-(15
    0,0)
390:LINE (0,0)-(0,
    -450)
400:L=450/M:N=N/15
    0
410:W=0:Q=F
420:FOR C=0TO M-1
422:COLOR 2:
    GLCURSOR (-50,
    W-15)
424:LPRINT USING "
    ####";Q
430:COLOR 1
435:G=INT (H(C)/N)
440:LINE (0,W)-(G,
    W)-(G,W-L)-(0,
    W-L)
450:W=W-L
470:Q=Q+B
480:NEXT C
490:END
```

PROGRAM TITLE 度数分布図 3

■フローチャート

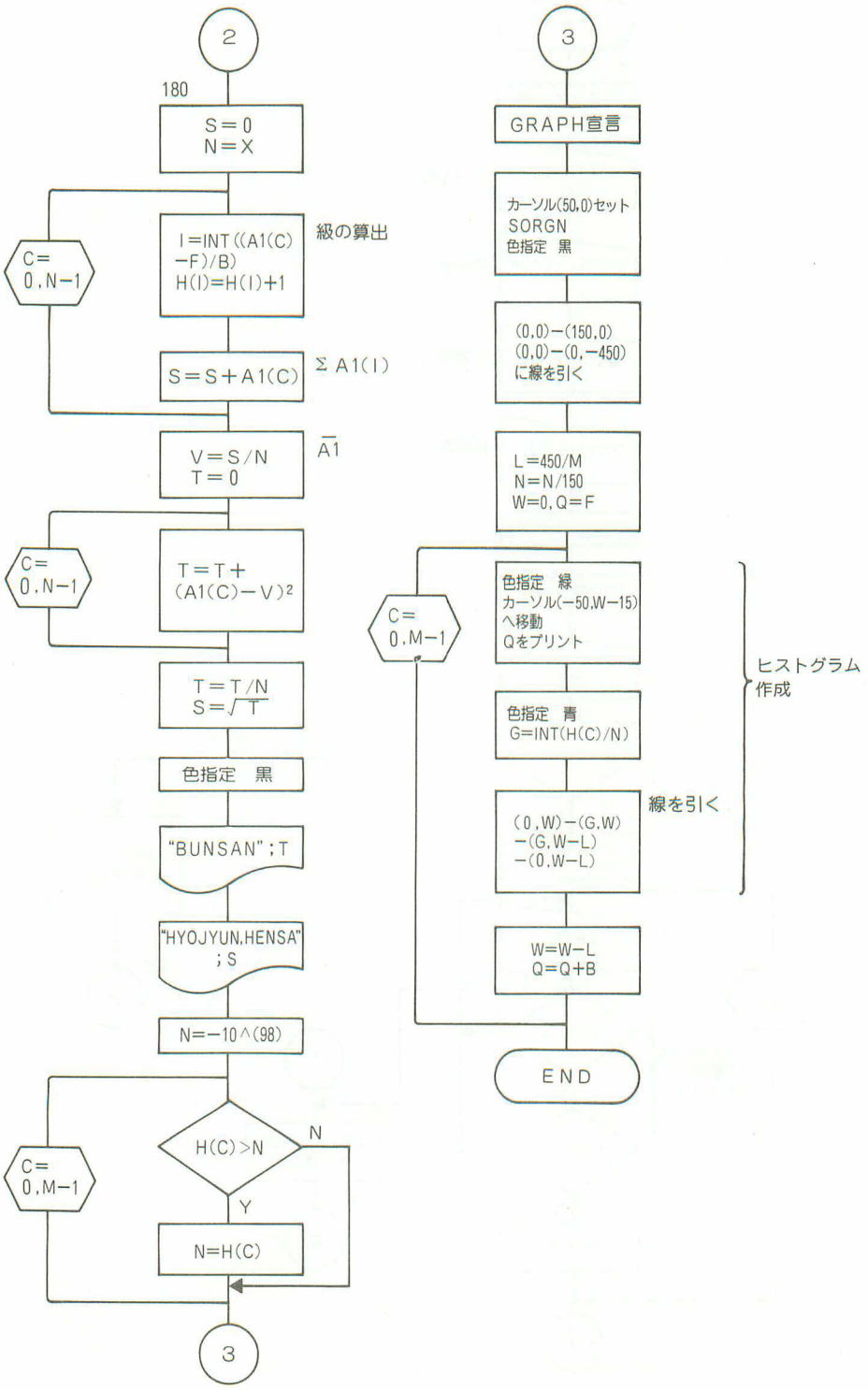


PROGRAM TITLE

度数分布図

3

■フローチャート



PROGRAM TITLE

度数分布図

4

■キー操作手順

ステップ	キー入力	表示	備考
1	DEF A	DATA−SU=	
2	10 ENTER	SHOKI−CHI=	
3	0 ENTER	KUKAN−HABA=	
4	2 ENTER	KUKAN−SU=	
5	5 ENTER	KEY−IN?(Y, N)	Y：キーボードから N：カセットテープから
6	Y ENTER	DATA=	
7	5 ENTER	DATA=	
8	2 ENTER	DATA=	
9	⋮	⋮	

メモリー内容					
A		A$	√	A1(N-1)	データ・テーブル
B	区間幅	B$			
C	√	C$		H(M-1)	級データ・テーブル
F	初期値	F$			
G	√	G$			
I	級NO	I$			
L	√	L$			
M	区間数	M$			
N	データ数	N$			
Q	√	Q$			
S	$\Sigma A1(i)$、$\sqrt{T}$	S$			
T	$\frac{1}{N}\Sigma(A1(i)-\overline{A})^2$	T$			
V	平均値	V$			
W	√	W$			
X	有効データ数	X$			
Z	有効最大値	Z$			

PROGRAM
TITLE

バイオリズム

1

カラーグラフィック
プリンター(CE-150)必要

概要

今月のあなたの状態は？

心と体の状態は健康のバロメーター。その日のあなたの運勢をも左右する大きな原因になります。そこで未然に自分のバイオリズムがわかれば、とても便利ですね。あなたの名前と生年月日を入れると、あなたの希望年月の(1ヶ月分)バイオリズムグラフがプリントされ、身体(緑色)、感情(赤色)、知性(青色)の曲線があなたの運勢にアドバイスを与えてくれます。

■使い方

- まず希望する年月を入力します。次に氏名(16文字まで)と生年月日を入力します。(全て西暦で入力して下さい。)
- 希望年月のバイオリズムが、各要素毎に色を変えてプリントされます。

■例題

希望年月　1981年7月
氏　名　SHARP　} を入力します。
生年月日　1952年1月28日

■内容

入力：調査年月
　　名前
　　生年月日
出力：調査年月のバイオリズム曲線を色を変えてプリントします。
曲線の計算式
　　身体　X＝Sin((B＋Y)/23＊360)＊80
　　感情　X＝Sin((C＋Y)/28＊360)＊80
　　知性　X＝Sin((D＋Y)/33＊360)＊80
　　B、C、D：生年月日から調査年月までの通算日数を各周期で割った余りの日数
　　Y：日付
　　(＋)、(－)方向に各々16㎜をMAXとしています。
周期　身体(23日)、感情(28日)、知性(33日)

PROGRAM
TITLE

バイオリズム

2

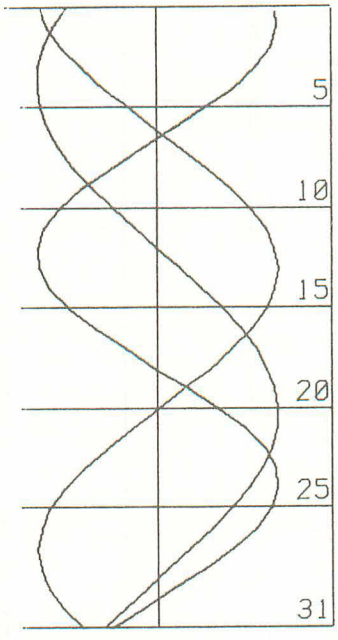
■プリント出力　※文字の大きさは18字／行

（実際はカラープリント出力です。
P.121参照）

```
DATE 1981,  7
NAME  SHARP
BIRTH 1952,  1,  28

-- SHINTAI
-- KANJYO
-- CHISEI

(-)        (+)
```



■キー操作手順

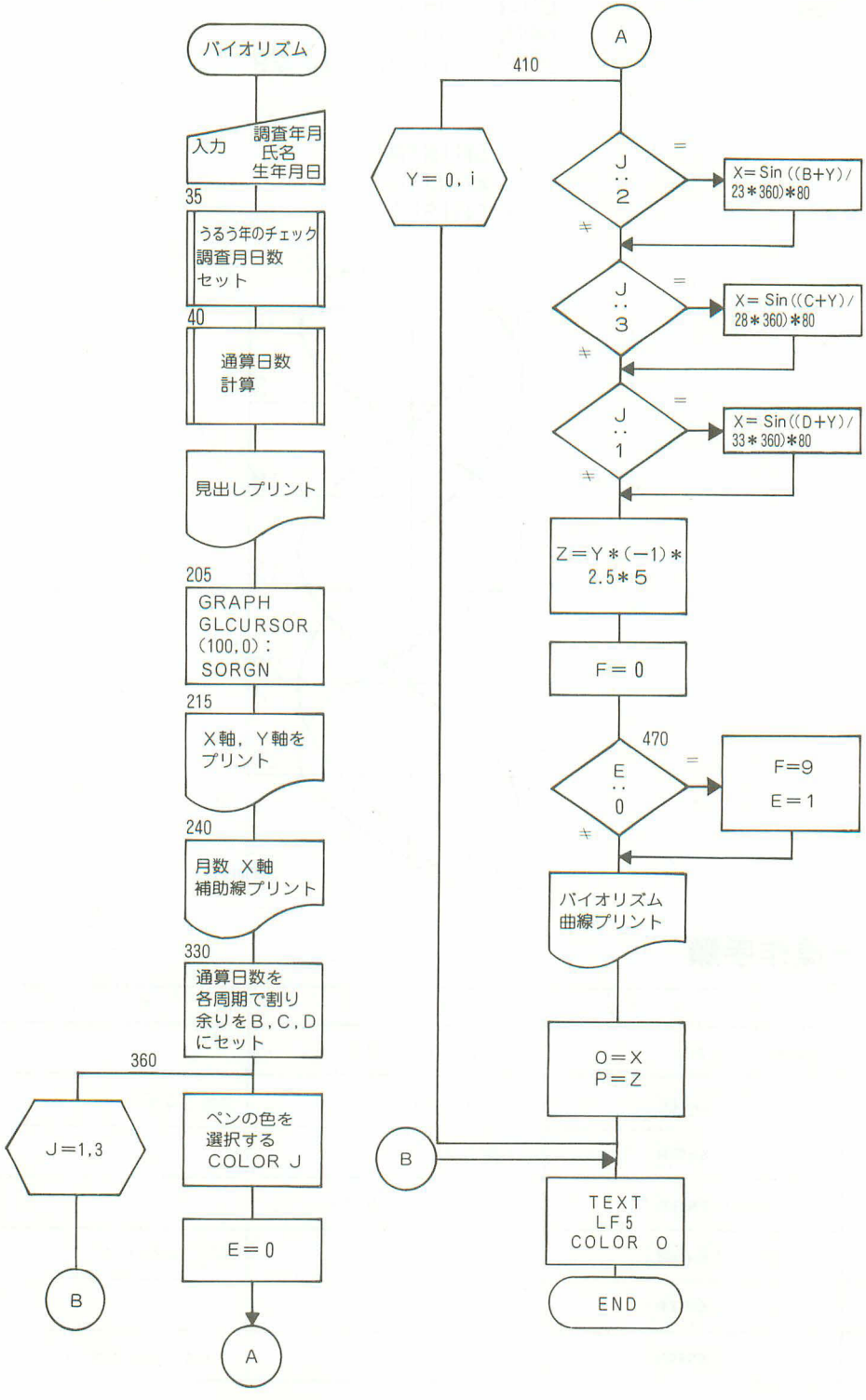
ステップ	キー入力	表示	備考
1	DEF　A	DATE?YEAR=	
2	1981　ENTER	MONTH=	西暦で4桁入力
3	7　ENTER	NAME?	
4	SHARP　ENTER	BIRTH?YEAR=	
5	1952　ENTER	MONTH=	西暦で4桁入力
6	1　ENTER	DAY=	
7	28　ENTER	>	バイオリズム曲線出力

PROGRAM TITLE

バイオリズム

3

■フローチャート

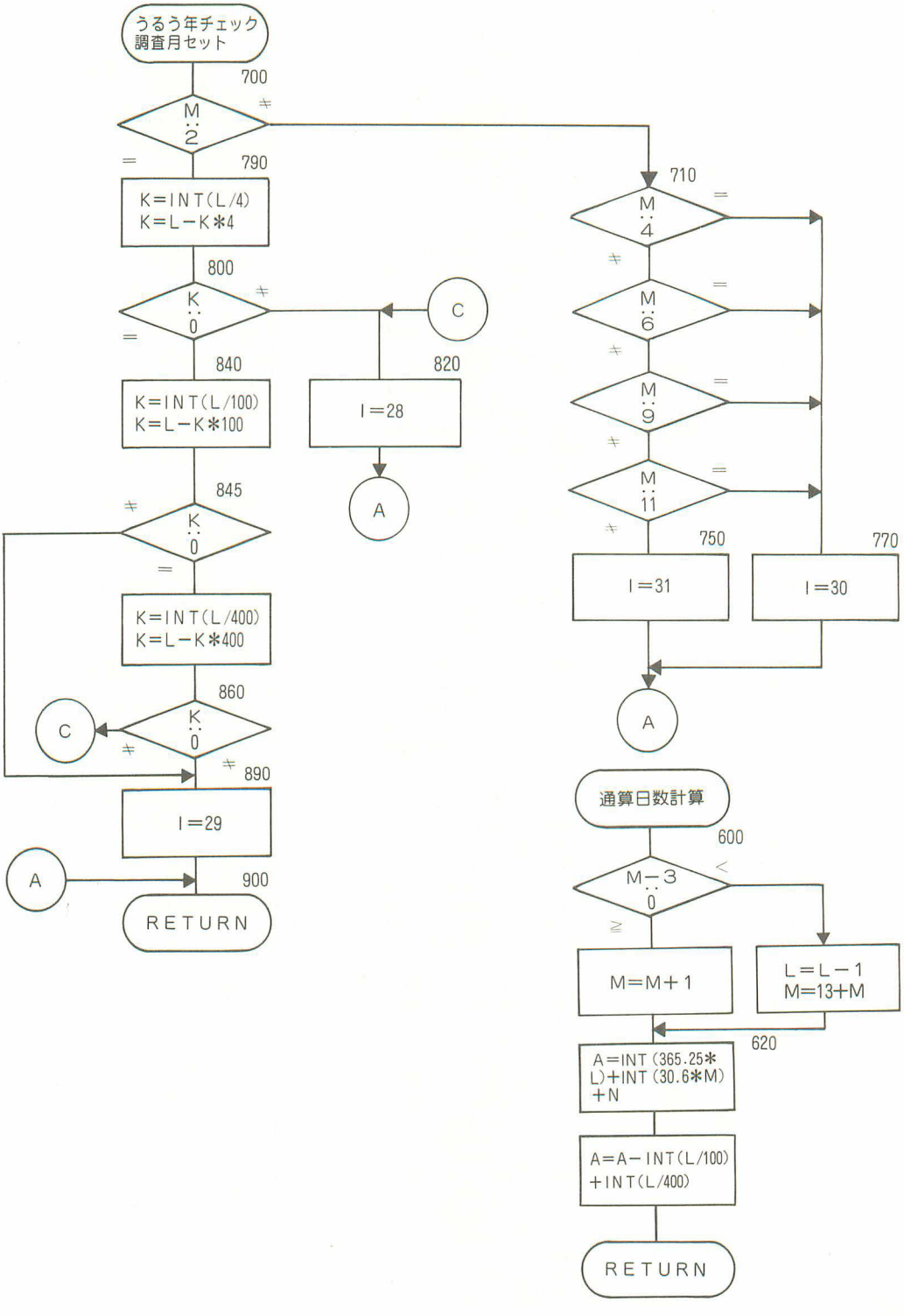


PROGRAM TITLE

バイオリズム

3

■フローチャート



PROGRAM
TITLE

バイオリズム

4

■プログラムリスト ※文字の大きさは18字／行

```
 10:"A":CLEAR :
    INPUT "DATE?
    YEAR=";L,"MONT
    H=";M
 15:TEXT :COLOR 0
 20:LPRINT "DATE";
    USING "#####";
    L;",";USING "#
    ##";M
 30:N=0
 35:GOSUB 700
 40:GOSUB 600:O=A
 50:INPUT "NAME? "
    ;A$
 60:LPRINT "NAME "
    ;A$
 70:INPUT "BIRTH?
    YEAR=";L,"MONT
    H=";M,"DAY=";N
 80:LPRINT "BIRTH"
    ;USING "#####"
    ;L;",";USING "
    ###";M;",";
    USING "###";N
 90:GOSUB 600:P=A
100:A=O-P:O=0:P=0
110:LF 2
120:COLOR 2
130:LPRINT " -- SH
    INTAI"
140:COLOR 3
150:LPRINT " -- KA
    NJYO"
160:COLOR 1
170:LPRINT " -- CH
    ISEI"
180:LF 1
190:COLOR 0
200:LPRINT "  (-)
         (+)"
205:GRAPH
210:GLCURSOR (100,
    0):SORGN
215:Y=I*2.5*5*(-1)
220:LINE (-100,0)-
    (115,0)
230:LINE (0,0)-(0,
    Y)
235:LINE (115,Y)-(
    115,0)
240:FOR Q=5TO 30
    STEP 5
243:R=Q
245:IF Q=30LET R=I
250:Y=R*2.5*(-1)*5
260:LINE (-90,Y)-(
    115,Y)
270:X=80
290:Z=Y+5
300:LINE (115,Z)-(
    X,Z),9
310:LPRINT R
320:NEXT Q
330:B=INT (A/23):B
    =A-(23*B)
340:C=INT (A/28):C
    =A-(28*C)
350:D=INT (A/33):D
    =A-(33*D)
360:FOR J=1TO 3
395:COLOR J
400:E=0
410:FOR Y=0TO I
420:IF J=2LET X=
    SIN ((B+Y)/23*
    360)*80
430:IF J=3LET X=
    SIN ((C+Y)/28*
    360)*80
440:IF J=1LET X=
    SIN ((D+Y)/33*
    360)*80
450:Z=Y*(-1)*2.5*5
460:F=0
470:IF E=0LET F=9:
    LET E=1
480:LINE (O,P)-(X,
    Z),F
490:O=X:P=Z
500:NEXT Y
510:NEXT J
515:TEXT :LF 5:
    COLOR 0
520:END
600:IF M-3>=0LET M
    =M+1:GOTO 620
610:L=L-1:M=13+M
620:A=INT (365.25*
    L)+INT (30.6*M
    )+N
625:A=A-INT (L/100
    )+INT (L/400)
630:RETURN
640:END
700:IF M=2GOTO 790
710:IF M=4GOTO 770
720:IF M=6GOTO 770
730:IF M=9GOTO 770
740:IF M=11GOTO 77
    0
750:I=31:GOTO 900
770:I=30:GOTO 900
790:K=INT (L/4):K=
    L-K*4
800:IF K=0GOTO 840
820:I=28:GOTO 900
840:K=INT (L/100):
    K=L-K*100
845:IF K=0GOTO 850
847:GOTO 890
850:K=INT (L/400):
    K=L-K*400
860:IF K=0GOTO 890
870:GOTO 820
890:I=29
900:RETURN
910:END
```

メモリー内容	
A	生年月日から調査年月日までの通算日数
B	通算日数を周期で割った余りをセット（身体）
C	〃 （感情）
D	〃 （知性）
E	1回目のジャッジ
F	PEN−UP，DOWNのコード
G	
H	
I	該当月　日数
J	ループカウンタ
K	うるう年計算ＷＯＲＫ
L	生年　調査年
M	生月　調査月
N	生日
O	西暦0年から調査年月日までの日数
P	西暦0年から生年月日までの日数
Q	ループカウンタ
R	見出し用日数ＷＯＲＫ
S	
T	
U	
V	
W	
X	バイオリズム曲線×座標
Y	1ヵ月の日数カウンタ
Z	バイオリズム曲線Ｙ座標
A$	氏名

PROGRAM TITLE

タイプ練習

1

概要

クイックキーオペレーション！

君はキーボードをどれだけ速く、正確に押せるか。

このプログラムで練習して、キーオペレーションをぐっと楽にしておけばプログラムの打ち込みも迅速処理。自己の可能性を高めよう。

■使い方（遊び方）

ブザーが鳴ってから3～6文字のアルファベットが表示されますから、それと同じものを制限時間内にキーボードから入れます。

全て合っていれば10点、半数以上合っていれば5点が合計得点に加算されます。制限時間を越えると次の問題が出て来ます。制限時間はグレード（1、2、3）と、表示される文字数により決まりますが、グレードの1が最も制限時間が短く、3が最も長くなっています。

同じグレード内で得点を競います。

問題は10問、最高得点は100点です。

■内容（計算内容など）

文字数（3～6）は乱数により決めます。

文字列（A～Z）も乱数により抽出します。

時間はグレードと文字数によって決まります。

PROGRAM TITLE

タイプ練習

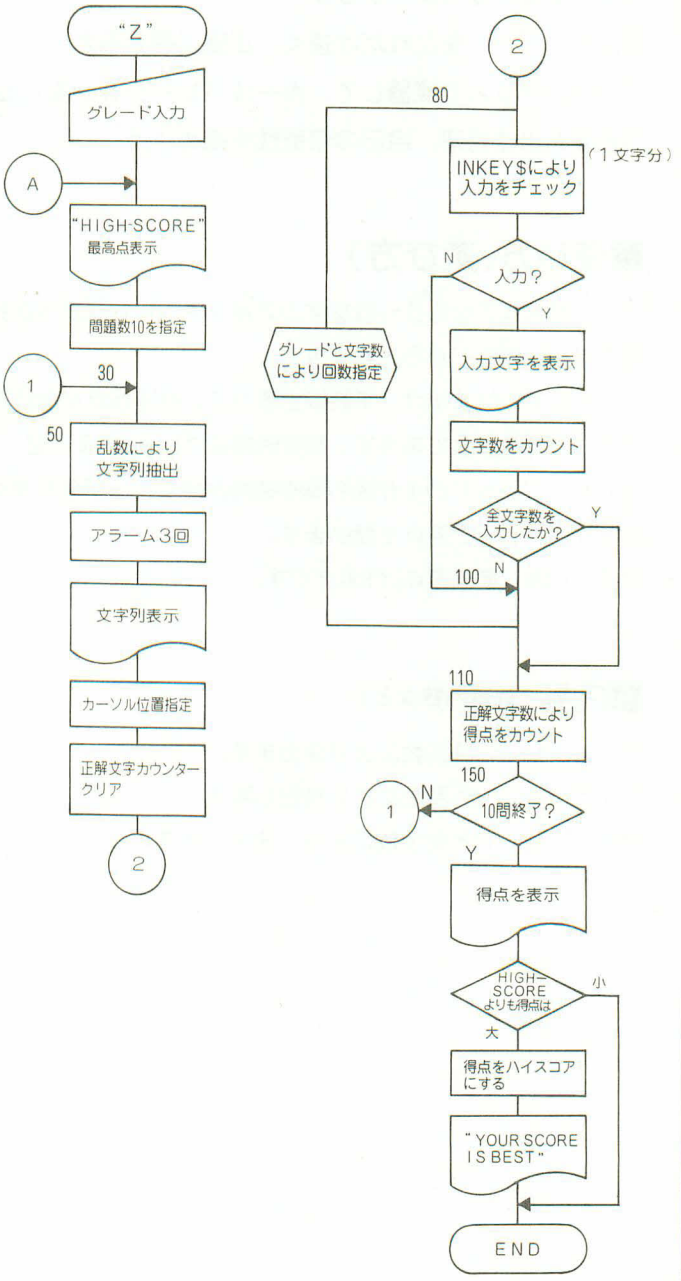
2

■プログラムリスト

※文字の大きさは18字／行

```
 10:"Z":CLEAR :CLS
    :DIM A$(5),B$(
    5):RANDOM
 15:INPUT "GRADE(1
    ,2,3)?";L
 17:IF (L=1)+(L=2)
    +(L=3)<>1THEN
    15
 20:"A":WAIT 0:P=0
    :PAUSE "HIGH-S
    CORE=";X
 30:FOR S=1TO 10
 40:B=RND 4+2:Y$="
     ":R=INT (B/2)
 50:FOR C=0TO B-1:
    B$(C)=""
 60:D=RND 26:A$(C)
    =CHR$ (D+&40):
    Y$=Y$+CHR$ (D+
    &40):NEXT C
 70:CLS :BEEP 3:
    PRINT Y$:
    CURSOR 10:E=0
 80:FOR W=1TO B*10
    *L:B$(E)=
    INKEY$ :IF B$(
    E)=""THEN 100
 85:PRINT B$(E);
 90:E=E+1:IF E=B
    LET W=400
100:NEXT W:Q=0
110:FOR W=0TO B-1:
    IF A$(W)=B$(W)
    LET Q=Q+1
120:NEXT W:IF Q<=R
    THEN 150
130:IF Q=BLET P=P+
    10:GOTO 150
140:P=P+5
150:NEXT S:CLS :
    BEEP 3:PAUSE "
    YOUR-SCORE=";P
160:IF P>XLET X=P:
    PRINT "YOUR SC
    ORE IS BEST"
170:WAIT :PRINT :
    END
```

■フローチャート



PROGRAM TITLE

タイプ練習

3

■キー操作手順

ステップ	キー入力	表示	備考
1	DEF Z	GRADE(1, 2, 3)?	グレードの変更又は最初にこのゲームを行う時1,2,3のいずれかを ENTER する。
2	DEF A	HIGH-SCORE=XXX	DEF Z を入力の時は必要なし。
3		XXXXXX（6文字以内の文字列）	アラーム3回
4	++++++ステップ3で表示された文字列	XXXXXX ++++++入力された文字	全文字を入力するか、一定時間過ぎると問題がかわる。10回くり返す。
⋮	⋮	⋮	
		YOUR-SCORE=XXX	アラーム3回 得点が表示されます。
		YOUR SCORE IS BEST	最高点を超える得点の場合にのみ表示されます。

メモリー内容					
A		A$		A$(0~5)	乱数により発生させた文字
B	文字数(3~6)	B$		B$(0~5)	入力された文字
C	√	C$			
D	√	D$			
E	入力文字数	E$			
F		F$			
G		G$			
H		H$			
I		I$			
J		J$			
K		K$			
L	グレード	L$			
M		M$			
N		N$			
O		O$			
P	得点	P$			
Q	正解文字数	Q$			
R	√	R$			
S	問題数	S$			
T		T$			
U		U$			
V		V$			
W	時間	W$			
X	最高得点	X$			
Y		Y$	問題の文字列		
Z		Z$			

PROGRAM TITLE

ストップウオッチ，タイマー，目覚し時計

1

概要

このプログラムではＰＣ-１５００に内蔵の時計は、こんなにも便利だという例をお見せします。これ一台で、ストップウオッチもタイマーも目覚し時計もＯＫです。

■使い方（遊び方）

（Ａ）アラーム時刻（０～23時０～59分０～59秒）を入力しますと、予定の時刻にメロディで知らせてくれます。
この間、時刻も表示します。

（Ｄ）タイマー時間をセットして SPACE キーでスタートを入力すると指定時間経過後、メロディで知らせてくれます。
この間、経過時間も表示されます。

（Ｓ）ストップウオッチと同様に SPACE キーでスタートとエンドを入力します。この間、経過時間が表示されます。

（注意）このプログラムを使用する前に、内蔵時計（ＴＩＭＥ）を正しくあわせて下さい。

■例題

（Ａ）アラーム時刻10時57分00秒としますと、105700と入力します。
NOW-TIME 10H 54M 5S と現在時刻が表示されつづけ10時57分00秒になりますとメロディがなります。

（Ｄ）タイマー時間を０時間30分00秒としますと、003000と入力します。
SPACE を押しますと、KEIKA 0H 0M 1S のように経過時間が表示されつづけ、０時間30分00秒が経過すると、メロディがなります。

（Ｓ） SPACE を押しますと 0M 0S 2SS のイメージで経過時間が流れます。次に SPACE を押しますと、STOP 10M59S 4SS 経過時間が表示され、処理は終ります。

■内容（計算内容など）

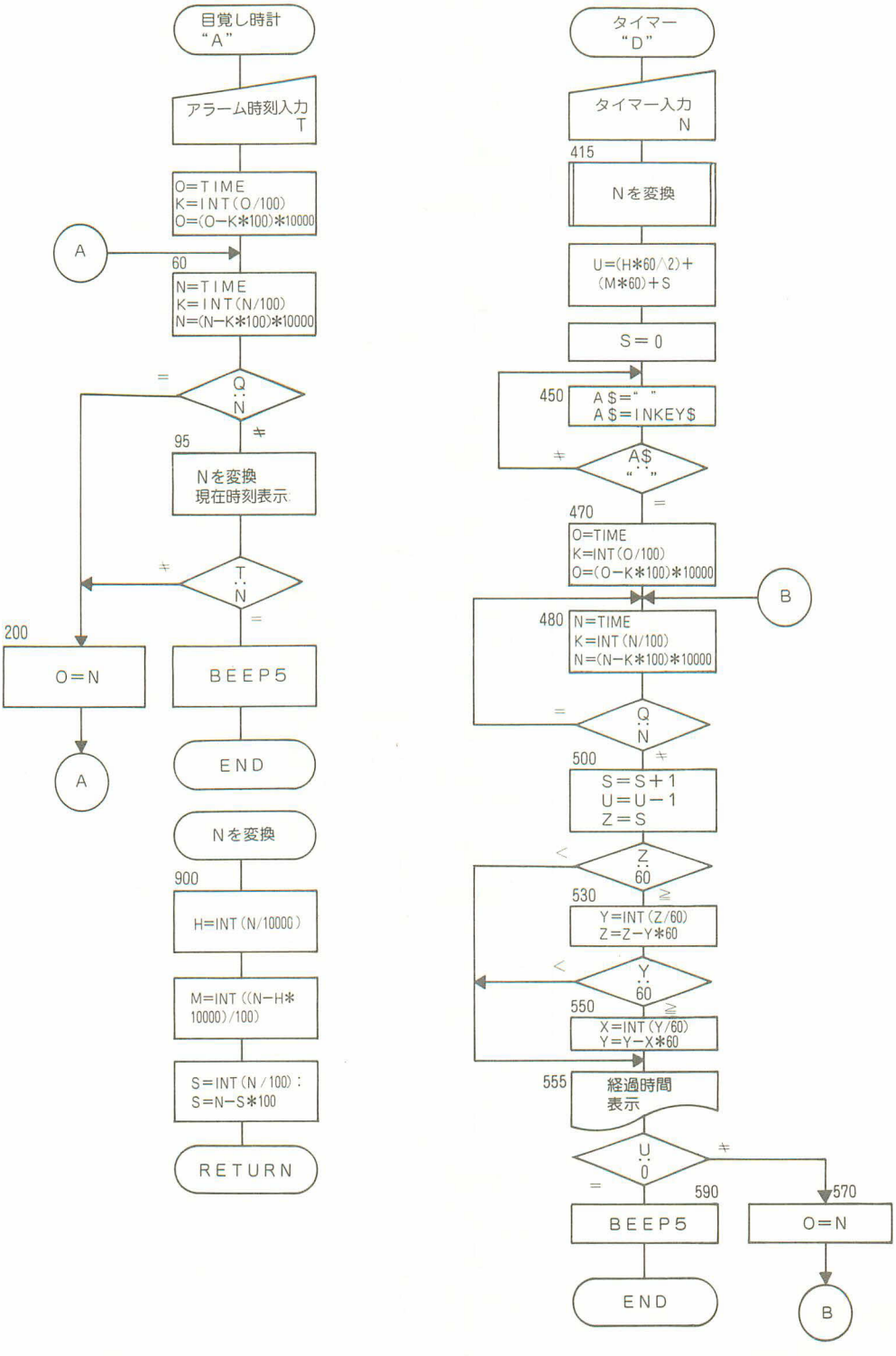
（Ａ）目覚し時計の働き
アラーム時刻（時、分、秒）を入力します。
予定時間になればメロディを発生して知らせます。 この間、その時の時刻を表示します。

（Ｄ）タイマーの働き
タイマー時間（時間、分、秒）を入力します。
SPACE でスタートして指定時間経過後、メロディを発生して知らせます。この間秒単位で経過時間を表示します。

（Ｓ）ストップウオッチの働き
SPACE でスタートして SPACE で終ります。この間1/6秒単位で経過時間を表示します。

PROGRAM TITLE　ストップウオッチ，タイマー，目覚し時計　2

■フローチャート

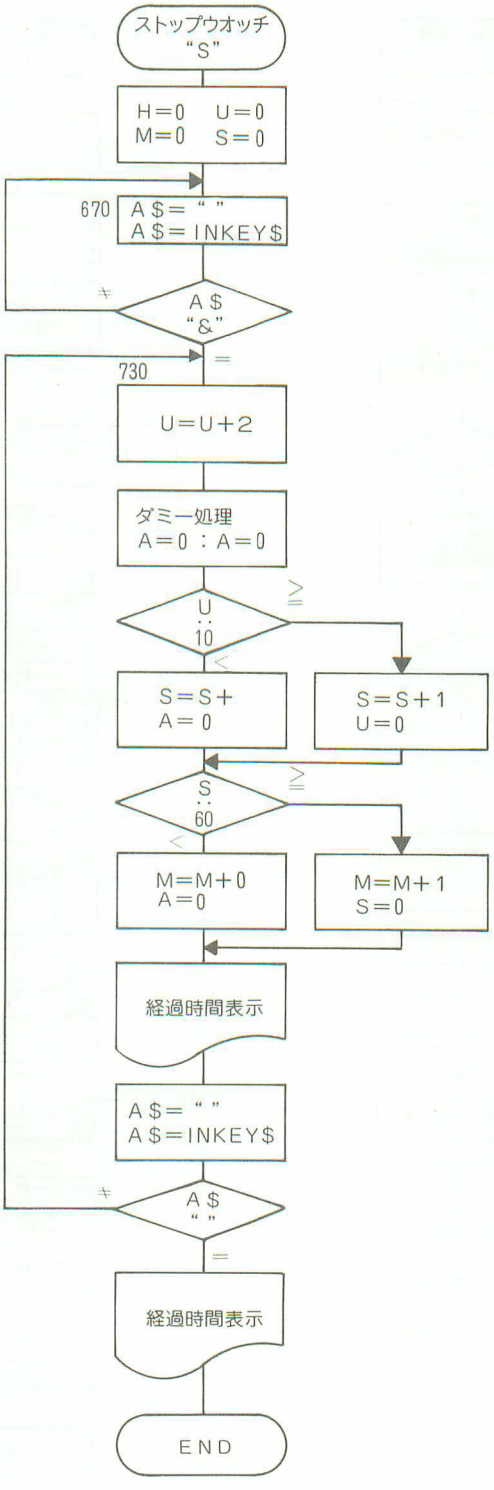


PROGRAM TITLE

ストップウオッチ，タイマー，目覚し時計

2

■フローチャート



PROGRAM TITLE

ストップウオッチ，タイマー，目覚し時計

3

■キー操作手順

ステップ	キ ー 入 力	表 示	備 考
1	TIME=101105350 ENTER	10110.535	現時刻を入力 1月1日10時53分50秒
2	DEF A	ALARM−TIME?	アラームタイム指定 10時57分00秒
3	105700 ENTER	NOW−TIME 10H53M56S	現時刻表示
4		NOW−TIME 10H57M00S	アラームが20回鳴る
1	DEF D	TIMER?	時間指定 00時30分00秒後セット
2	003000 ENTER	TIMER003000	
3	SPACE	KEIKA 0H 0M 1S	
4		KEIKA 0H 30M 0S	アラームが20回鳴る
1	DEF S	STOP WATCH	
2	SPACE	0M 0S 0SS	ストップウオッチスタート
		M S SS	
3	SPACE	STOP 1M 0S 0SS	1分後

PROGRAM TITLE

ストップウオッチ，タイマー，目覚し時計

4

■プログラムリスト ※文字の大きさは18字／行

```
 10:"A":CLEAR :
    WAIT 0
 20:INPUT "ALARM-T
    IME?";T
 30:O=TIME
 40:K=INT (O/100)
 50:O=(O-K*100)*10
    000
 60:N=TIME
 70:K=INT (N/100)
 80:N=(N-K*100)*10
    000
 90:IF O=NGOTO 200
 95:GOSUB 900
100:PRINT "NOW-TIM
    E";USING "###"
    ;H;"H";USING "
    ###";M;"M";
    USING "###";S;
    "S"
120:IF T=NBEEP 20:
    GOTO 300
200:O=N
210:GOTO 60
300:END
400:"D":CLEAR :
    WAIT 0
410:INPUT "TIMER?"
    ;N
415:GOSUB 900
420:U=(H*60^2)+(M*
    60)+S
440:S=0
445:A$=""
450:A$=INKEY$
460:IF A$<>" "GOTO
    445
470:O=TIME :K=INT
    (O/100):O=(O-K
    *100)*10000
480:N=TIME :K=INT
    (N/100):N=(N-K
    *100)*10000
490:IF O=NGOTO 480
500:S=S+1
505:U=U-1
510:Z=S
520:IF Z<60GOTO 55
    5
530:Y=INT (Z/60):Z
    =Z-Y*60
540:IF Y<60GOTO 55
    5
550:X=INT (Y/60):Y
    =Y-X*60
555:PRINT "KEIKA";
    USING "##";X;"
    H";USING "###"
    ;Y;"M";USING "
    ###";Z;"S"
560:IF U=0GOTO 590
570:O=N:GOTO 480
590:BEEP 20
600:END
650:"S":CLEAR :
    WAIT 0
655:PRINT "STOP WA
    TCH"
660:H=0:M=0:S=0:U=
    0
670:A$=""
680:A$=INKEY$
690:IF A$<>" "GOTO
    670
730:U=U+2
735:A=0:A=0
740:IF U<10LET S=S
    +0:A=0
750:IF U=10LET S=S
    +1:U=0
760:IF S<60LET M=M
    +0:A=0
770:IF S=60LET M=M
    +1:S=0
810:PRINT M;"M";
    USING "###";S;
    "S";USING "##"
    ;U;"SS"
815:A$=""
820:A$=INKEY$
830:IF A$<>" "GOTO
    730
840:WAIT :USING :
    PRINT "STOP";M
    ;"M";S;"S";U;"
    SS"
850:END
900:H=INT (N/10000
    )
910:M=INT ((N-H*10
    000)/100)
920:S=INT (N/100):
    S=N-S*100
930:RETURN
940:END
```

PROGRAM TITLE

ストップウオッチ，タイマー，目覚し時計

5

メモリー内容（ストップウオッチ）	
A	WORK
B	
C	
D	
E	
F	
G	
H	経過時間（時間）
I	
J	
K	
L	
M	経過時間（分）
N	
O	
P	
Q	
R	
S	経過時間（秒）
T	
U	経過時間（1/10秒）
V	経過時間
W	
X	
Y	
Z	
A$	INKEY$

メモリー内容（タイマー）	
A	
B	
C	
D	
E	
F	
G	
H	タイマー時間（時）
I	
J	
K	計算WORK
L	
M	タイマー時間（分）
N	タイマー時間：経過時間（NOW）
O	経過時間（OLD）
P	
Q	
R	
S	タイマー時間（秒）
T	
U	タイマー時間を秒と換算
V	
W	
X	経過時間（時）
Y	経過時間（分）
Z	経過時間（秒）
A$	INKEY$

メモリー内容（目覚し時計）	
A	
B	
C	
D	
E	
F	
G	
H	現在時刻（時）
I	
J	
K	計算WORK
L	
M	現在時刻（分）
N	経過時間（NOW）
O	経過時間（OLD）
P	
Q	
R	
S	現在時刻（秒）
T	アラーム時間
U	
V	
W	
X	
Y	
Z	

PROGRAM TITLE

コンピュータ・フラワー

1

カラーグラフィックプリンター(CE-150)必要

概要

カラーグラフィックプリンタを用いて、いろいろなデザインが作れるようになれれば非常に楽しいですね。それでは、このプログラムで、花の図柄を描いて見ましょう。

■使い方(遊び方)

DEF A を押せば、かわいい花が描き出されます。

■内容(計算内容など)

X(ｉ)＝Sin(６＊ｉ)＊Cos(ｉ＋A)＊80

Y(ｉ)＝Sin(６＊ｉ)＊Sin(ｉ＋A)＊80

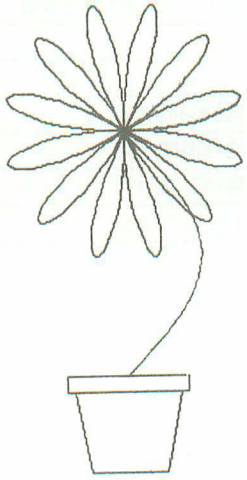
１つの花びらに対して上記ｉの値を１～30まで変化させ30座標を線分で結びます。

Aの値を0°～330°まで30°づつ12回変化させることにより、12枚の花びらの変化した座標を求めます。

DEGREEモードで(DEGが表示されていることを確認して)使用してください。

■プリント出力

(実際はカラープリント出力です。P.121参照)



■キー操作手順

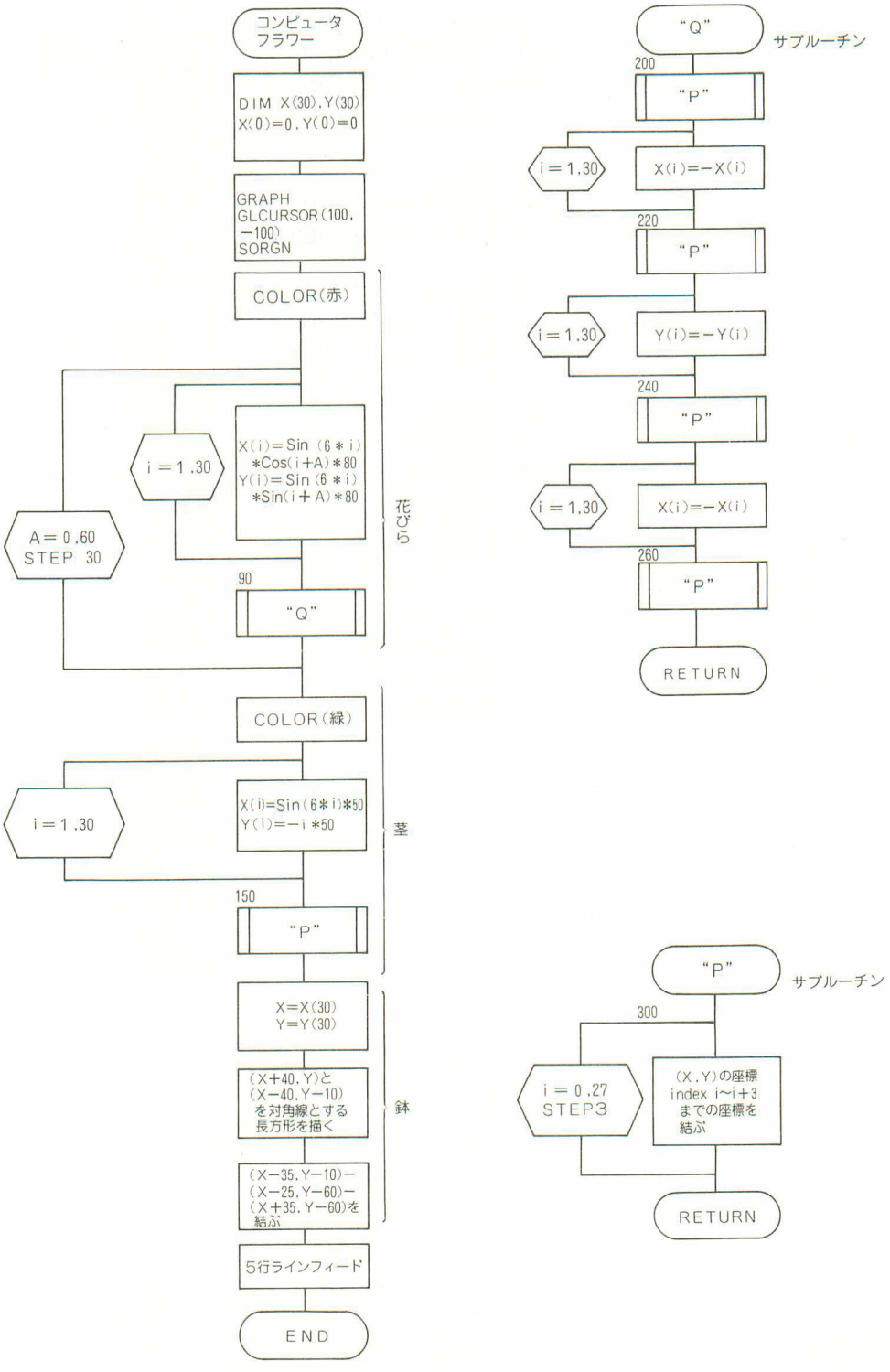
ステップ	キー入力	表示	備考
1	DEF A	>	花の図柄出力

PROGRAM TITLE

コンピュータ・フラワー

2

■フローチャート



PROGRAM TITLE

コンピュータ・フラワー

3

■プログラムリスト ※文字の大きさは18字／行

```
 10:"A":CLEAR :DIM
    X(30),Y(30):X(
    0)=0:Y(0)=0
 20:GRAPH
 30:GLCURSOR (100,
    -100):SORGN
 35:COLOR 3
 40:FOR A=0TO 60
    STEP 30
 50:FOR I=1TO 30
 60:X(I)=SIN (6*I)
    *COS (I+A)*80
 70:Y(I)=SIN (6*I)
    *SIN (I+A)*80
 80:NEXT I
 90:GOSUB "Q"
100:NEXT A
105:COLOR 2
110:FOR I=1TO 30
120:X(I)=SIN (6*I)
    *50
130:Y(I)=-I*5
140:NEXT I
150:GOSUB "P"
155:X=X(30):Y=Y(30
    )
160:LINE (X+40,Y)-
    (X-40,Y-10),0,
    0,B
170:LINE (X-35,Y-1
    0)-(X-25,Y-60)
    -(X+25,Y-60)-(
    X+35,Y-10)
180:TEXT :LF 5:END
200:"Q"GOSUB "P"
210:FOR I=0TO 30:X
    (I)=-X(I):NEXT
    I
220:GOSUB "P"
230:FOR I=0TO 30:Y
    (I)=-Y(I):NEXT
    I
240:GOSUB "P"
250:FOR I=0TO 30:X
    (I)=-X(I):NEXT
    I
260:GOSUB "P"
270:RETURN
300:"P"FOR I=0TO 2
    7STEP 3
310:LINE (X(I),Y(I
    ))-(X(I+1),Y(I
    +1))-(X(I+2),Y
    (I+2))-(X(I+3)
    ,Y(I+3))
320:NEXT I
330:RETURN
```

メモリー内容	
A	ＦＯＲ文の変数
B	✓
C	
D	
E	
F	
G	
H	
I	ＦＯＲ文の変数
J	
K	
L	
M	
N	
O	
P	
Q	
R	
S	
T	
U	
V	
W	
X	X(30)の置き換え
Y	Y(30)の置き換え
Z	
X(30)	X座標
Y(30)	Y座標

PROGRAM TITLE

コンピュータ・グラフィック

1

カラーグラフィック
プリンター(CE-150)必要

概要

幾何学模様を描いて、変化の度合を知りたい時にはこんなプログラムを作ってみるのも一興でしょう。連続した正三角形の列は、アンモナイトの生長過程のようにも見えます。

■使い方(遊び方)

角度、増分、個数を様々なパターンで入力することによって描かれる模様の変化をお楽しみください。

■例題

(例1) 個数30
角度10
増分3.5
} を入力した場合の作画(プリント出力参照)

(例2) 個数35
角度20
増分3
} を入力した場合の作画

■内容(計算内容など)

R=R+K
(Rは増分の累計で初期値5, 入力増分Kの値をワンパターン各に足し込みます。)
T=T+S
(Tは角度の累計で初期値−S 入力角度Sの値をワンパターン各に足し込みます。)
X1=R*SinT
Y1=R*CosT
X2=R*Sin(T+60)
Y2=R*Cos(T+60)
(0、0)−(X1、Y1)−(X2、Y2)−(0、0)を直線で結びます。

上記のパターンを入力個数:N回繰り返し作画します。

DEGREEモードで(DEGが表示されていることを確認して)使用してください。

PROGRAM TITLE

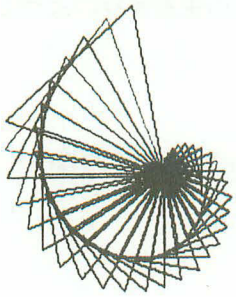
コンピュータ・グラフィック

2

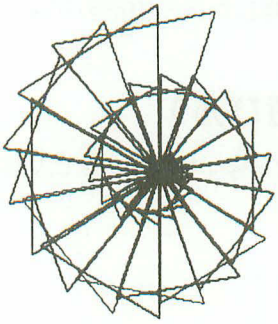
■プリント出力

（実際はカラープリント出力です。P.121参照）

（例1）



（例2）



■プログラムリスト ※文字の大きさは18字／行

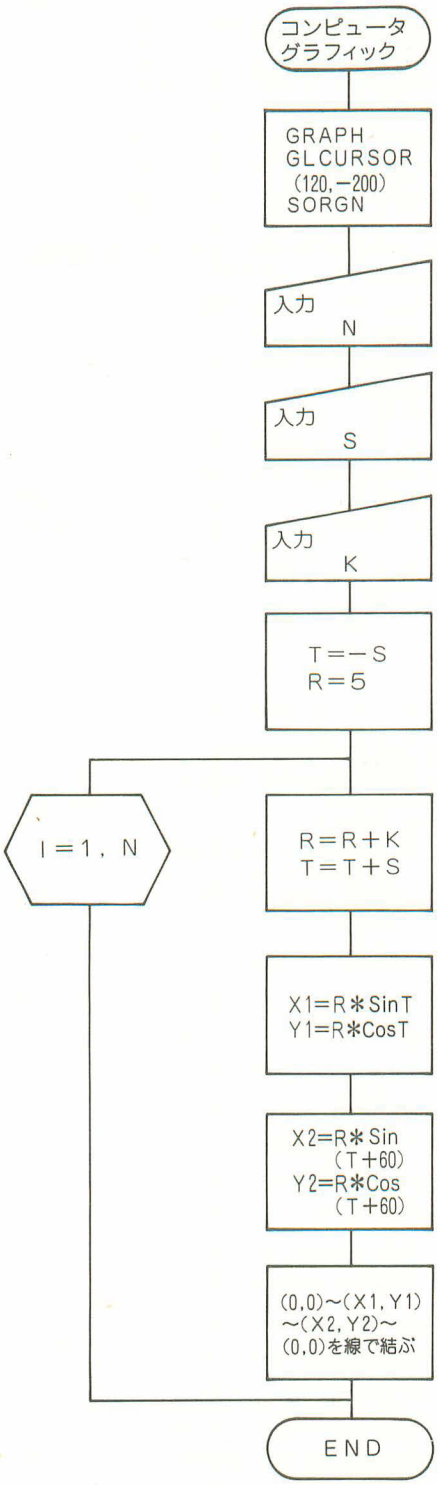
```
 10:"A":GRAPH :
    RANDOM
 20:GLCURSOR (120,
    -200):SORGN
 30:INPUT "KOSU=";
    N
 40:INPUT "KAKUDO=
    ";S
 50:INPUT "ZOUBUN=
    ";K
 60:T=-S:R=5
 70:FOR I=1TO N
 75:COLOR (RND 4-1
    )
 80:R=R+K:T=T+S
 90:X1=R*SIN T:Y1=
    R*COS T
100:X2=R*SIN (T+60
    ):Y2=R*COS (T+
    60)
110:LINE (0,0)-(X1
    ,Y1)-(X2,Y2)-(
    0,0)
120:NEXT I
130:END
```

PROGRAM TITLE

コンピュータ・グラフィック

3

■フローチャート



N：個数(入力)
S：角度(入力)
K：増分(入力)
T：角度
S：増分

PROGRAM TITLE

コンピュータ・グラフィック

4

■キー操作手順

ステップ	キー入力	表示	備考
1	DEF A	KOSU=	
2	30 ENTER	KAKUDO=	
3	10 ENTER	ZOUBUN=	
4	3.5 ENTER	>	
1	DEF A	KOSU=	
2	35 ENTER	KAKUDO=	
3	20 ENTER	ZOUBUN=	
4	3 ENTER	>	

メモリー内容			
A		X1	作画X座標の1
B		Y1	〃 Y 〃 の1
C		X2	〃 X 〃 の2
D		Y2	〃 Y 〃 の2
E			
F			
G			
H			
I	√		
J			
K	増分(INPUT)		
L			
M			
N	個数(INPUT)		
O			
P			
Q			
R	増分(計算値)		
S	角度(INPUT)		
T	角度(計算値)		

エラーコードとエラー内容

エラーコード	エ　ラ　ー　内　容
1	文法エラー。 プログラムでのみ実行可能な命令をマニュアル操作で、マニュアル操作でのみ実行可能な命令をプログラムで実行しようとした。
2	ＮＥＸＴ命令に対するＦＯＲ命令がない。 ＲＥＴＵＲＮ命令に対するＧＯＳＵＢ命令がない。
4	ＲＥＡＤ命令に対するＤＡＴＡがない。
5	すでに宣言されている配列変数名を再度宣言している。
6	配列変数をＤＩＭ命令で宣言せずに使っている。
7	変数名が不適合。（数値変数に文字を代入しようとした場合など）。
8	配列変数の指定が３次元以上になっている。
9	配列変数の添字がＤＩＭ命令で宣言した配列の大きさを越えている。
10	変数をメモリー上に確保できない。
11	GOSUB命令、THEN命令などで指定されたラインがプログラム上に存在しない。
12	ＵＳＩＮＧ命令におけるフォーマット指定エラー。（指定の形が正しくない）
13	プログラムエリアの容量オーバーあるいはリザーブエリアの容量オーバー。
14	ＦＯＲ－ＮＥＸＴ命令の段数が深く、スタックエリアの容量をオーバーする。 データや計算命令（ファンクション）のスタック容量オーバー。
15	ＧＯＳＵＢ命令の段数が深く、スタックエリアの容量をオーバーする。 文字列の加算や大小比較において、その文字合計が80バイトを越える。
16	指数部が99を超える数値が指定されている。（10Ｅ99など） ＆ＦＦＦＦ を超える16進数値が指定されている。
17	式においてデータの形式が不適合（１＋"Ａ"、Ａ＄＊２など）
18	演算式においてデータ数が正しくない。（ＳＩＮ(30，60)など）
19	指定の数値が規定されている範囲から外れている（ＤＩＭ　Ａ(256)など）
20	固定変数を@（　）　あるいは@＄（　）　で指定するとき、（　）の指定がない。
21	変数でなければならないところが、数式などで指定されている。
22	プログラムを計算機に転送するとき、転送するエリアがない。
23	時刻の設定エラー。
26	指定されているモードで実行できない命令を実行しようとした。
27	指定した命令、関数がない。（オプション機器を本体から切り離した場合に生じることがあります。）
28	“　　”ではさまれた中に命令や関数のコードを挿入している。あるいはＩＮＰＵＴ命令やＡＲＥＡＤ命令にて文字変数に命令や関数のコードを代入しようとした。
30	プログラム入力時、ラインナンバーが 65539 を超えている。 （65280～65539はＥＲＲＯＲ　１）
32	ＩＮＰＵＴ命令が実行されているとき、ＧＣＵＲＳＯＲ命令で152～155ポイントが指定されていて、キー入力した内容を表示する桁がない。

エラーコード	エ　ラ　ー　内　容
36	ＵＳＩＮＧ命令で指定されたフォーマットにしたがってデータあるいは文字を表示できない。
37	計算結果が9.99999999999Ｅ99よりも大きい（オーバーフローエラー）
38	除数が０の除算を実行。
39	不合理な演算をおこなった。（ＬＮ－30、ＡＳＮ1.5など）
131	文字列および文字変数の直前に符号と見なされる＋およびーがついている
177〜181	プログラム入力により、すでに定義されていたデータ領域が破壊された。
0 224〜241	ＩＮＰＵＴ命令，ＲＥＡＤ命令あるいはＡＲＥＡＤ命令実行時に入力データの終了がＢＡＳＩＣ命令などになっている。 ＤＡＴＡ命令に続くデータの区切りが，（コンマ）になっていない。 また、データの終了が：（コロン）あるいは ENTER になっていない。
テープレコーダーに関するエラー	
40	式の指定が不適切
42	記録されているプログラムが大きすぎて、計算機のプログラムエリアに転送し切れない。
43	ＣＬＯＡＤ？で照合内容の不一致が発生した。 データの転送において、ＩＮＰＵＴ＃命令で指定された変数の形と記録されているデータの形が一致しない。
44	チェックサムエラーが発生した。
プリンタに関するエラー	
70	ペンが －2048≦Ｘ，Ｙ≦2047 の座標範囲を越える。
71	ＴＥＸＴモードで10.24㎝以上紙を逆に進めようとした。
72	ＴＡＢ命令で指定した値が不適当。
73	ＧＲＡＰＨモードで使用できない命令、あるいはＴＥＸＴモードで使用できない命令を実行しようとした。
74	ＬＩＮＥあるいはＲＬＩＮＥ命令で（，）で指定する座標の数が多い。
76	TEXTモードにおいて、ＬＰＲＩＮＴ命令で計算結果（数値）が１行で印字できない。
78	次の理由により、ＬＰＲＩＮＴ、ＬＩＮＥなどの命令が実行できない。 ① ペン交換中 ② ローバッテリーチェック（ＥＲＲＯＲ 80）により掛ったプリンタロックが解除されていない。
79	カラー信号が出ていない。
80	ローバッテリーチェック（プリンタの内部電池の電圧が低下し、プリンタが正しく働かない恐れがあるときに、プリンタをロックします。ＣＥ-150の取扱説明書を参照してください。）

エラー1が出た場合以下のことに留意してください。

(1) 入力に関して

◎ 入力ミスを訂正するような場合

10 APRINT A$ [ENTER]

↓

10 GPRINT A$ [ENTER]

↑ AをGに変更した

◎ CURSOR命令をGCURSOR命令に変更するような場合

10 CURSOR 10 [ENTER]

↓

10 GCURSOR 10 [ENTER]

↑ Gを挿入した

◎ GPRINT 入力時 [G] [DEF] [W] と操作したような場合

以上のような方法によって修正あるいは入力された場合、計算機には1つの命令として判断されません。このような場合は、その命令のスペルをアルファベットキーにより最初から再入力していただく必要があります。

（例）

	10：APRINT A$
	┌ カーソル点滅
[►]	10 APRINT A$
[SHIFT] [DEL ◄]	10 PRINT A$
[SHIFT] [DEL ◄]	10 A$
[SHIFT] [INS ►] ····· [SHIFT] [INS ►]	10 □□□□□ A$
[G] [P] [R] [I] [N] [T] [ENTER]	10：GPRINT A$

なお、命令が正しく入力されているかどうかはカーソルキーでチェックすることができます。

（正しい入力の場合）

┌ カーソル点滅

GPRINT A$

↓

[►] GPRINT A$

（間違った入力の場合）

GPRINT A$

↓

[►] GPRINT A$

↓

[►] GPRINT A$

キャラクタ・コード表

下位ビット(下位桁)↓ ＼ 上位ビット(上位桁)→	0000	0001	0010	0011	0100	0101	0110	0111
16進	0	1	2	3	4	5	6	7
0000	0			0	@	P		p
0001	1		!	1	A	Q	a	q
0010	2		"	2	B	R	b	r
0011	3		#	3	C	S	c	s
0100	4		$	4	D	T	d	t
0101	5		%	5	E	U	e	u
0110	6		&	6	F	V	f	v
0111	7		'	7	G	W	g	w
1000	8		(	8	H	X	h	x
1001	9		)	9	I	Y	i	y
1010	A		*	:	J	Z	j	z
1011	B		+	;	K	[	k	{
1100	C		,	<	L	¥	l	¦
1101	D		−	=	M	]	m	}
1110	E		.	>	N	^	n	~
1111	F		/	?	O	_	o	■

注）太ワクの記号は、表示をおこなうことはできますが、プリンタではプリントされません。

関数・命令(省略形)一覧

1．基本関数

関　数	省略形	備　考
SIN	SI.	sin
COS		cos
TAN	TA.	tan
ASN	AS.	$\sin^{-1}$
ACS	AC.	$\cos^{-1}$
ATN	AT.	$\tan^{-1}$
LOG	LO.	$\log_{10} X$
LN		$\log_e X$
EXP	EX.	e^x
^		べき乗
√（SQR）	SQ.	開平
DEG		度分秒→10進数度変換
DMS	DM.	10進数度→度分秒変換
INT		整数化（インテジャー）　INT(10/3)　ENTER　→ 3
ABS	AB.	絶対値（アブソリュート）ABS(−5＊2)　ENTER　→10
SGN	SG.	符号関数（シグナム）
π（PI）		円周率を指定（π≒3.141592654）

2．BASIC命令

（※……プログラムで実行可能な命令
★……マニュアル操作で実行可能な命令）

	命　令	省略形	一　般　形	備　考
※	LET （レット）	LE.	(1) LET 数値変数＝式 (2) LET 文字変数＝“文字” (3) LET 文字変数＝文字変数	LETはIF文に続く場合以外は省略可
※	INPUT （インプット）	I. 〜 INPU.	(1) INPUT 変数，変数，… (2) INPUT “文字”，変数，“文字”，変数，… (3) INPUT “文字”；変数，“文字”；変数，…	入力命令 データを入力
※ ★	PRINT （プリント）	P. 〜 PRIN.	(1) PRINT { 式 / “文字” / 文字変数 } (2) PRINT { 式 / “文字” / 文字変数 } , { 式 / “文字” / 文字変数 } (3) PRINT { 式 / “文字” / 文字変数 } ; { “文字” / 変数 } ; …… … ; { “文字” / 変数 } (4) 上記一般形(1)，(3)の最後に；をつけた形	出力命令 指定された内容を表示します。
※	END （エンド）	E. EN.	END	プログラム終了命令
※ ★	DEGREE （ディグリー）	DE. 〜 DEGRE.	DEGREE	角度単位指定 “度”単位指定〔°〕
※ ★	RADIAN （ラディアン）	RAD. 〜 RADIA.	RADIAN	角度単位指定 “ラディアン”単位指定〔rad〕
※ ★	GRAD （グラード）	GR. GRA.	GRAD	角度単位指定 “グラード”単位指定〔g〕
※ ★	GOTO （ゴートゥ）	G. 〜 GOT.	(1) GOTO { 式 / “文字” / 文字変数 } (2) GOTO （単独の場合はマニュアル操作でのみ実行可能）	ジャンプ命令 （プログラム実行開始命令） ・指定されたラインあるいはラベルへ実行を移す。 ・指定されたラインからプログラムを実行する。

	命　令	省略形	一　般　形	備　考
※	IF (イフ)		IF { 条件式 / 式 } 実行文	判断命令 与えられた条件を判断し、実行を次の実行文か、あるいは次のラインへ移します。
※	THEN (ゼン)	T. 〜 THE.	IF文における実行文として定義します。 THEN { 式 / "文字" / 文字変数 }	ジャンプ命令 IF文における実行文として定義することのみ可能。
※	FOR (フォー) TO (トゥー) STEP (ステップ)	F. FO. STE.	(1) FOR 数値変数＝式$_1$ TO 式$_2$ (2) FOR 数値変数＝式$_1$ TO 式$_2$ STEP 式$_3$ 式$_1$：初期値，　式$_2$：最終値 式$_3$：増（減）分	FOR－NEXTループの始まり NEXT命令と対で使用
※	NEXT (ネクスト)	N. 〜 NEX.	NEXT 数値変数 数値変数はFOR命令に続く数値変数と一致していなければならない。	FOR－NEXTループの末端を示します。 FOR文と対で使用。
※	GOSUB (ゴーサブ)	GOS. GOSU.	GOSUB { 式 / "文字" / 文字変数 }	サブルーチンジャンプ命令 指定されたラインあるいはラベルへ実行を移し、そこからのプログラムをサブルーチンとして実行。 RETURN命令と対で使用。
※	RETURN (リターン)	RE. 〜 RETUR.	RETURN	復帰命令 サブルーチンの復帰命令で、GOSUBの次のステートメントへ実行が復帰します。
※ ★	DIM (ディメンジョン)	D. DI.	(1) DIM 変数名(式) (2) DIM 変数名(式)＊式$_3$ (3) DIM 変数名(式$_1$，式$_2$) (4) DIM 変数名(式$_1$，式$_2$)＊式$_3$ 変数名：A、B、C$、D$ など (　)：配列の大きさ、次元指定 式$_3$：桁数指定（文字変数のみ）	変数の定義 配列変数の変数名、大きさなどを定義し、プログラム・データエリアに確保する。

	命　令	省略形	一　般　形	備　考
※ ★	USING (ユージング)	U. ～ USIN.	(1) USING "＃…＃．＃…＃＾" (2) USING "＆＆＆……＆＆＆" (3) PRINT USING"フォーマット";… (4) USING (5) PRINT USING ;…	フォーマット指定命令 数値データ、文字データの出力フォーマット指定 フォーマット指定解除
※ ★	CURSOR (カーソル)	CU. ～ CURSO.	(1) CURSOR 式 (式の値は 0～25) (2) CURSOR	表示開始位置指定 式の値で表示ポジションを指定 指定を解除。
※ ★	WAIT (ウェイト)	W. ～ WAI.	(1) WAIT 式 (式の値は 0～65535) (2) WAIT	表示時間指定 PRINT命令によるプログラム停止時間を指定。 指定を解除。 (停止時間無限)
※	PAUSE (ポーズ)	PA. ～ PAUS.	一般形はPRINT命令と同じ形をとります。	出力命令 指定された内容を約0.85秒表示したのちプログラムを実行。
※ ★	CLS (クリアーズ)		CLS	表示クリア命令 表示を消去します。
※ ★	GCURSOR (グラフィックカーソル)	GC. ～ GCURSO.	GCURSOR 式 (式の値は 0～155 あるいは ＆0～＆9B)	表示開始位置指定 表示ポジションをドットの単位で指定。
※ ★	GPRINT (グラフィックプリント)	GP. ～ GPRIN.	(1) GPRINT "○○　○○　○○…" ("　" の中は16進数字) (2) GPRINT　○;○;… (3) GPRINT　＆○;＆○;…	出力命令 ドットパターンで指定された内容を表示します。
※	READ (リード)	REA.	READ 変数，変数，…	データの読み込み命令 DATA命令に続いて指定されているデータを指定の変数に入れる。

	命　令	省略形	一　般　形	備　考
※	DATA (データ)	DA. DAT.	DATA 式, 式, …	データ指定 READ命令に続く変数に与えるデータを指定。
※	RESTORE (リストア)	RES. 〜 RESTOR.	(1) RESTORE 式 (2) RESTORE	READ命令により読み込まれるデータの順番を変える命令
※	ON GOTO (オン ゴートゥー)	ONG. O. G. 〜 O. GOT.	ON 式 GOTO 式$_1$, 式$_2$, 式$_3$, …	ジャンプ命令 ONに続く式の値によりジャンプ先を指定する。
※	ON GOSUB (オン ゴーサブ)	ONGOS. O. GOS. O. GOSU.	ON式 GOSUB 式$_1$, 式$_2$, 式$_3$, …	サブルーチンジャンプ命令 ONに続く式の値によりジャンプ先を指定。
※	ON ERROR (オン エラー)	ONER. O. ER. 〜 O. ERRO.	ON ERROR (ON ERROR GOTO 式)	エラー処理のための命令
※	REM (リマーク)		REM 注釈	プログラムの中に非実行文（注釈）を指定するための命令。
※	ARUN (オートラン)	ARU.	ARUN	電源オン時、自動的にプログラムの実行を開始する命令。
※	STOP (ストップ)	S. 〜 STO.	STOP	プログラムの実行を停止する命令。
※	AREAD (オートリード)	A. 〜 AREA.	AREAD 変数	定義付けキーによるプログラム実行開始時に、表示内容を指定された変数に入れる。
※ ★	CLEAR (クリア)	CL. 〜 CLEA.	CLEAR	データ（変数）をすべて消去する命令。
※ ★	BEEP (ビープ)	B. 〜 BEE.	BEEP 式$_1$, 式$_2$, 式$_3$ BEEP ON BEEP OFF	音命令 音の高さや、長さを指定。 音発生機能をオン 音発生機能をオフ

	命　令	省略形	一　般　形	備　考
※★	TR ON (トレース オン)	TR. TRO.	TR ON	デバッグをおこなうモードを指定。
※★	TR OFF (トレース オフ)	TROF.	TR OFF	デバッグをおこなうモードを解除。
※★	LOCK (ロック)	LOC.	LOCK	MODE キーで切替えるモードをロックする。
※★	UNLOCK (アンロック)	UN. 〜 UNLOC.	UNLOCK	モードのロックを解除する。
※★	RANDOM (ランダ マイズ)	RA. 〜 RANDO.	RANDOM	RND命令に先だって乱数のタネを植えつけるものです。
※★	OPN (オープン)	OP.	OPN OPN〝デバイス名〟	CE-158で使用

3．コマンド(マニュアル操作でのみ実行可能)

	命　令	省略形	一　般　形	備　考
★	NEW (ニュー)		(1) NEW (2) NEW 0(PROモードでのみ実行可能) (3) NEW 式(PROモードでのみ実行可能) 式の値－1のアドレスまでを機械語エリアとし、それ以降のプログラムエリアおよび全変数をクリアする。(式≠0)	PROモードでプログラムおよび全変数をクリア。 RESERVE モードでリザーブ内容をクリア。
★	RUN (ラン)	R. RU.	(1) RUN (2) RUN 式 (3) RUN{"文字" / 文字変数}	プログラム実行開始命令。 RUN モードで有効
★	LIST (リスト)	L. 〜 LIS.	一般形はRUN命令と同様に定義	プログラムのリストをおこなう命令。 PROモードで有効
★	CONT (コンティ ニュー)	C. 〜 CON.	CONT	一時停止しているプログラムの実行を再開する命令。 RUN モードで有効

4. BASIC関数

	命　令	省略形	一　般　形	備　考
※★	RND（ランダム）	RN.	RND 式	乱数を得る命令。
※★	AND（アンド）	AN.	式 AND 式	論理積
※★	OR（オア）		式 OR 式	論理和
※★	NOT（ノット）	NO.	NOT 式	否　定
※★	LEFT$（レフトドル）	LEF. LEFT.	LEFT$（文字変数，数式）	指定された文字列の左側から、指定された桁数の文字を取り出す。
※★	RIGHT$（ライトドル）	RI. 〜 RIGHT.	RIGHT$（文字変数，数式）	指定された文字列の右側から、指定された桁数の文字を取り出す。
※★	MID$（ミッドドル）	MI. MID.	MID$（文字変数，数式$_1$，，数式$_2$）	指定された文字列の中間の文字を取り出す。
※★	LEN（レングス）		LEN{“文字”／文字変数}	指定された文字列の文字数を求めます。
※★	STR$（ストリングドル）	STR.	STR$ 数式 （A$＝STR$ A）	数値を文字列に変換
※★	VAL（バリュー）	V. VA.	VAL{“文字”／文字変数}	文字列を数値に変換
※★	ASC（アスキー）		ASC{“文字”／文字変数}	文字などのキャラクタをアスキーコードに変換。
※★	CHR$（キャラクタドル）	CH. CHR.	CHR$ 数式	アスキーコードを文字などのキャラクタに変換。
※★	STATUS（ステータス）	STA. 〜 STATU.	STATUS 0 STATUS 1	プログラム・データエリアのプログラム以外のバイト数、プログラムのバイト数などの呼び出し。
※★	MEM（メム）	M. ME.	MEM	STATUS 0と同じプログラム・データエリアのプログラム以外のバイト数を表示。

	命　令	省略形	一　般　形	備　考
※ ★	POINT （ポイント）	POI. POIN.	POINT 式 （式の値は 0～155 あるいは &0～&9B） 例　A＝POINT 56	ドットパターン読み出し命令 指定されたポイントに表示されているドットパターン情報を読み出す。

5．その他

	命　令	省略形	一　般　形	備　考
※ ★	TIME （タイム）	TI. TIM.	(1) TIME＝月日時．分秒 (2) TIME	タイム関数 月日時．分秒をセット。 日付、時刻呼び出し
※ ★	INKEY$ （イン キードル）	INK. ～ INKEY.	文字変数＝INKEY$	INKEY$ 命令実行時にキーが押されていればその内容を読み込む。

6．プリンタ関係命令

	命　令	省略形	一　般　形	備　考
※ ★	TEXT （テキスト）	TEX.	TEXT	TEXTモード指定 文字や数字を印字するモード。
※ ★	GRAPH （グラフィック）	GRAP.	GRAPH	GRAPHモード指定 グラフなど図形を書くモード。
※ ★	LPRINT （ライン プリント）	LP. ～ LPRIN.	一般形はPRINT命令と同じ形をとります。	出力命令 指定された内容を印字します。
※ ★	CSIZE （キャラクタ サイズ）	CSI. CSIZ.	CSIZE 式 （式の値は 1～9）	印字する文字の大きさを指定。
※ ★	ROTATE （ローテイト）	RO. ～ ROTAT.	GRAPHモードでのみ有効。 ROTATE 式 （式の値は 0～3）	印字する文字の向き（印字方向）を指定。
※ ★	COLOR （カラー）	COL. COLO.	COLOR 式 （式の値は 0～3）	文字などの色を指定

	命　令	省略形	一　般　形	備　考
※★	TEST (テスト)	TE. TES.	TEST	色のチェック TEST命令実行で5㎜角の正方形をそれぞれの色で書く。
※★	TAB (タブ)		TEXTモードでのみ有効。 (1) TAB 式 (2) LPRINT TAB 式；…	ペンの位置を指定
※★	LCURSOR	LC. ～ LCURSO.	TABと同じ形	TABと同じ
※★	LF (ライン フィード)		TEXTモードでのみ有効。 LF 式	紙送り命令 式で示す行数分の紙送りをおこないます。
※★	GLCURSOR (グラフィックラインカーソル)	GL. ～ GLCURSO.	GRAPHモードでのみ有効。 GLCURSOR(式$_1$，式$_2$)	ペン位置を原点からX軸、Y軸方向に移動させる命令。
※★	SORGN (セット オリジン)	SO. ～ SORG.	GRAPHモードでのみ有効。 SORGN	現在あるペンの位置を新しい原点に指定する命令。
※★	LINE (ライン)	LIN.	GRAPHモードでのみ有効。 (1) LINE(式$_1$，式$_2$)－(式$_3$，式$_4$) (2) LINE(式$_1$，式$_2$)－(式$_3$，式$_4$)，式$_5$，式$_6$ (3) LINE(式$_1$，式$_2$)－(式$_3$，式$_4$)，式$_5$，式$_6$，B 式$_5$：線の種類を指定 式$_6$：色を指定 B ：四角形を指定 (4) LINE(式$_1$，式$_2$)－(式$_3$，式$_4$)－…(式$_{11}$，式$_{12}$)	線を書く命令です。
※★	RLINE (レラティブライン)	RL. ～ RLIN.	GRAPHモードでのみ有効。 一般形はLINE命令と同じ形となります。	ペン位置を原点として、線を書く命令です。
★	LLIST (ライン リスト)	LL. ～ LLIS.	(1) LLIST ENTER (2) LLIST式 ENTER (3) LLIST式$_1$，式$_2$ ENTER (4) 式$_1$あるいは式$_2$を省略した形	リスト命令 プリンタによりプログラムのリストアウトをおこなう命令。

7．テープレコーダーに関する命令

	命　令	省略形	一　般　形	備　考
※★	CSAVE (カセット セイブ)	CS. ～ CSAV.	(1) CSAVE (CSAVE-1) (2) CSAVE “ファイル名” (CSAVE-1“ファイル名”)	記録命令 プログラムやリザーブ内容をテープに記録する命令。

	命　令	省略形	一　般　形	備　考
★	CLOAD (カセット ロード)	CLO. CLOA.	(1) CLOAD (CLOAD-1) (2) CLOAD“ファイル名” (CLOAD-1“ファイル名”)	転送命令 プログラムあるいはリザーブ内容をテープから計算機に転送する命令。
★	MERGE (マージ)	MER. MERG.	一般形はCLOAD命令と同じ形となります。 〔この命令では、それまで記憶していたプログラムはそのまま保持し、その後に新しく読み込んだプログラムが記憶される。〕	転送命令 プログラムをテープから計算機に転送する命令。
※	CHAIN (チェイン)	CHA. CHAI.	(1) CHAIN “ファイル名” (CHAIN-1“ファイル名”) (2) CHAIN “ファイル名”, 式 (CHAIN-1“ファイル名”, 式) (3) 上記(1), (2)の“ファイル名”を省略した形 ((2)の形でファイル名を省略したとき、式がラベル指定の場合は(1)と同じ形式となります。)	転送命令 プログラムの中で使用する命令。プログラムをテープから読み込んで(転送して)そのプログラムを実行。
★	CLOAD? (カセット ロード?)	CLO.? CLOA.?	(1) CLOAD? (CLOAD?-1) (2) CLOAD?“ファイル名” (CLOAD?-1“ファイル名”)	照合命令 計算機内のプログラムあるいはリザーブ内容とテープに記録されている内容との照合をおこなう命令。
※ ★	PRINT# (プリント クロスハッチ)、	P.# 〃 PRIN.#	(1) PRINT# 変数名, 変数名, … (PRINT#-1, 変数名, 変数名, …) (2) PRINT#“ファイル名”; 変数名, 変数名, … (PRINT#-1,“ファイル名”; 変数名, 変数名, …)	データ記録命令 計算機に記憶されているデータをテープに記録する命令。
※ ★	INPUT# (インプット クロスハッチ)	I.# 〃 INPU.#	一般形はPRINT#命令と同じ形になります。	データ転送命令 テープに記録されているデータを指定された変数に転送する命令。
※ ★	RMT OFF (リモート オフ)	RM. OF.	RMT OFF	REM1端子のリモート機能を解除する命令。
※ ★	RMT ON (リモート オン)	RM. O.	RMT ON	REM1端子のリモート機能をふたたびセットする命令。

索　引

（ページ）

アスキーコード……73
一次元配列……43
エラー……15、26
エラーコード……167
エラー処理……26、58
オートパワーオフ……9
オールリセットスイッチ……4
音発生機能……62
カーソル……14
角度指定……16
記録……96
クリア……46
原点……88
サブルーチン……37
ジャンプ……29
条件式……30
照合……101
シンボル……115
数値変数……18、42
スタックエリア……41
ステートメント……22
定義付け……39
定義付けキー……40
データ専用エリア……45
デバッグ……68
転送……97
二次元配列……43
バイト……5、41、46
配列変数……42
判断……30
否定……67
表示ポジション……50
フォーマット指定……48
プレイバック……14
フローチャート……33
プログラミング……20
プログラムモード……10
プログラム・データエリア……45
プロンプト記号……9

（ページ）

変数……18、42
メインルーチン……37
メモリー……18、44
モード……10
文字変数……18、42
文字列……74
ライン……21
ラインナンバー……22
ラベル……30
乱数……60
ランモード……10
リザーブエリア……45
リザーブ機能……76
リザーブモード……10
リスト……25、92
論理積……64
論理和……64
ABS……16
ACS……16
AND……64、66
AREAD……58
ARUN……59
ASC……73、74
ASN……16
ATN……16
BEEP……62
CHAIN……99
CHR$……74
CLEAR……46
CLOAO……97
CLOAD?……101
CLS……51
COLOR……86
CONT……63
COS……16
CPU……5
CSAVE……96
CSIZE……85
CURSOR……50
DATA……55
DEG……16

（ページ）

DEGREE…… 16
DIM…… 42
DMS…… 16

END…… 22
EXP…… 16

FOR…… 35

GCURSOR…… 51
GLCURSOR…… 88
GOSUB…… 37
GOTO…… 29
GPRINT…… 52
GRAD…… 16
GRAPH…… 84

IF…… 30
INKEY$…… 61
INPUT……22、26
INPUT#……102
INT…… 16

LEFT$…… 70
LEN…… 72
LET…… 22
LF…… 87
LINE…… 89
LIST…… 25
LLIST…… 92
LN…… 16
LOCK…… 67
LOG…… 16
LPRINT…… 84

MEM…… 63
MERGE…… 98
MID$…… 71

NEW……4、23
NEXT…… 35
NOT……65、67

ON ERROR…… 58
ON GOSUB…… 57
ON GOTO…… 57
OPN…… 68
OR……64、66

PAUSE……50、51
PI…… 16
POINT…… 54

（ページ）

PRINT……22、27、47
PRINT#……102
RADIAN……16
RAM…… 5
RANDOM……60
READ……55
REM……58
RESTORE……55
RETURN……37
RIGHT$……70
RLINE……91
RMT OFF……104
RMT ON……104
RND……60
ROM…… 5
ROTATE……85
RUN……25

SGN…… 16
SIN…… 16
SORGN……88
SQR…… 16
STATUS……63
STEP……35
STOP……59
STR$……73

TAB……86
TAN…… 16
TEST……86
TEXT……84
THEN……31
TIME……61
TO……35

UNLOCK…… 67
USING…… 48

VAL…… 73

WAIT…… 50

SHARP

POCKET COMPUTER PC-1501 PROGRAM FORMAT PAGE1

PROGRAM T I T L E	※この欄にはなにも記載しないで下さい。

氏名	フリガナ	年令	電話	自宅		会員NO. (会員の方は必ず記入して下さい)
				勤務先	内線	
住所	フリガナ 〒□□□ - □□				勤務先	
					投稿年月日	年 月 日

■周辺機器：

■概　要：

■使い方：

PROGRAM TITLE	氏　名

■参　考：
（計算式など）

■例　題：

PROGRAM T I T L E	氏　名

■プリント出力(結果):

PROGRAM T I T L E	氏　名

■キー操作手順：

ステップ	キー入力	表　　示	備　　考

PROGRAM T I T L E	氏　名

■プログラムリスト：

ラインナンバー	ステートメント	ラインナンバー	ステートメント

PROGRAM TITLE	氏　名

■メモリー内容：

A		A$			
B		B$			
C		C$			
D		D$			
E		E$			
F		F$			
G		G$			
H		H$			
I		I$			
J		J$			
K		K$			
L		L$			
M		M$			
N		N$			
O		O$			
P		P$			
Q		Q$			
R		R$			
S		S$			
T		T$			
U		U$			
V		V$			
W		W$			
X		X$			
Y		Y$			
Z		Z$			

PROGRAM TITLE	氏　名

■フローチャート(流れ図)：

＊　＊　＊ＰＣ-1501とＰＣ-1500との違いについて＊　＊　＊

ＰＣ-1501はＰＣ-1500に比べ、ＲＡＭ容量が大きくなっています。
したがって「シャープポケットコンピュータＰＣ-1500機械語の手引」の内容のうち、チップセレクト回路、メモリーマップ、40ピンコネクタ（メモリーモジュール用）の記載の一部が異なります。
この異なる点についての概要を記載いたしますので、機械語を使用する際には十分留意してください。

１．チップセレクト回路

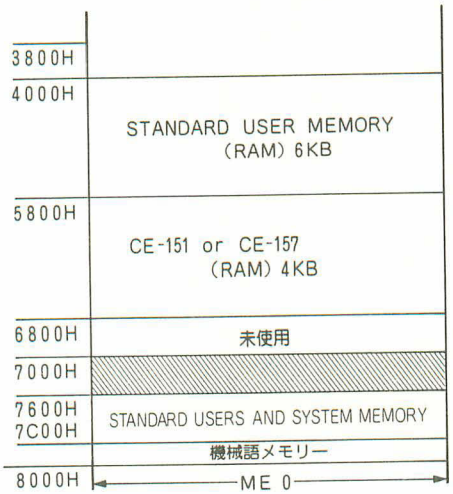
信号の取りかた（Ｓ０～Ｓ５）が変っています。(91ページ)

ＴＣ40Ｈ138Ｆの出力	メモリーアドレス	ＰＣ-1500	ＰＣ-1501
$\overline{Y0}$	４０００Ｈ ４７ＦＦＨ	S0 STANDARD USER MEMORY	S0 STANDARD USER MEMORY
$\overline{Y1}$	４８００Ｈ ４ＦＦＦＨ	S1	S1
$\overline{Y2}$	５０００Ｈ ５７ＦＦＨ	S2	S2
$\overline{Y3}$	５８００Ｈ ５ＦＦＦＨ	S3 OPTION USER MEMORY	S3 OPTION USER MEMOR
$\overline{Y4}$	６０００Ｈ ６７ＦＦＨ	S4	S4
$\overline{Y5}$	６８００Ｈ ６ＦＦＦＨ	S5	S5
$\overline{Y6}$	７０００Ｈ ７７ＦＦＨ	使用不可	使用不可
		7600Ｈ 7BFFＨ STANDARD USER AND SYSTEM MEMORY	7600Ｈ 7C00Ｈ STANDARD USER AND SYSTEM MEMORY
$\overline{Y7}$	７８００Ｈ ７ＦＦＦＨ	使用不可	機械語メモリー

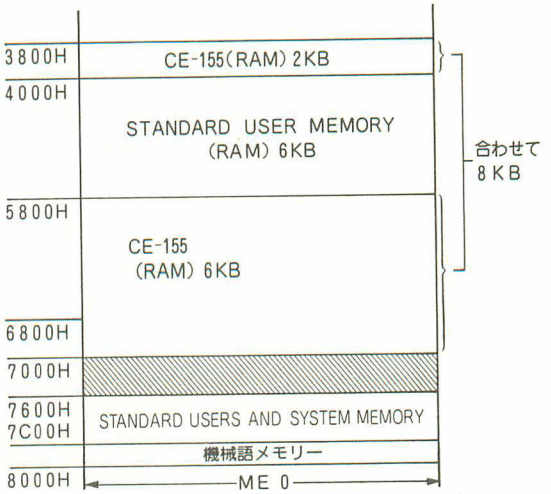
2．メモリーマップ

PC-1501にメモリーモジュールCE-151、CE-155、CE-157を装着したときのユーザーメモリーエリアが変ります。（93～94ページ）

CE-151またはCE-157使用時



CE-155使用時



3．40ピンコネクタ

メモリーモジュールを装着する40ピンコネクタの5及び16～18番ピンの信号と内容が変わっています。（100ページ）

ピンNo.	信号名	内　　容
5	S3	5800H～5FFFHのアドレス指定
16	S4	6000H～67FFHのアドレス指定
17	S5	6800H～6FFFHのアドレス指定
18		非接続

MEMO

MEMO

MEMO _____

保　証　書（保証規定）

本書は、本書記載内容で無料修理をさせて頂くことをお約束するものです。
保証期間中に故障が発生した場合は、製品と本書をご持参、ご提示の上、お買上げの販売店にご依頼ください。
お買上げ年月日、販売店名など記入もれがありますと無効となります。必ずご確認いただき、記入のない場合はお買上げの販売店にお申し出ください。
ご転居・ご贈答品などでお買上げ販売店に修理をご依頼できない場合は、製品に同梱しております「シャープサービス・お客様ご相談窓口一覧表」をご覧の上、もよりのサービス会社へご持参ください。
本書は再発行いたしません。たいせつに保管してください。

〈無料修理規定〉

1．取扱説明書・本体注意ラベルなどの注意書にしたがった正常な使用状態で、保証期間内に故障した場合には、お買上げ販売店、または当社サービス会社が無料修理いたします。

2．保証期間内でも、次の場合には有料修理となります。
（イ）本書のご提示がない場合。
（ロ）本書にお買上げ年月日・お客様名・販売店名の記入がない場合、または字句を書き換えられた場合。
（ハ）使用上の誤り、または不当な修理や改造による故障・損傷。
（ニ）お買上げ後に落とされた場合などによる故障・損傷。
（ホ）火災・公害および地震・風水害その他天災地変など、外部に要因がある故障・損傷。
（ヘ）電池の液もれによる故障・損傷。
（ト）消耗品（乾電池）が損耗し取り替えを要する場合。

3．本書は日本国内においてのみ有効です。
（THIS WARRANTY CARD IS ONLY VALID FOR SERVICE IN JAPAN.）

★この保証書は本書に明示した期間・条件のもとにおいて無料修理をお約束するものです。したがいましてこの保証書によってお客様の法律上の権利を制限するものではありませんので、保証期間経過後の修理などにつきましておわかりにならない場合はお買上げの販売店、またはシャープお客様ご相談窓口にお問い合せください。

修理メモ

シャープ計算機保証書
(WARRANTY CARD)

品　　名　計　算　機

形　　名　PC−1501

保証期間　お買上げ日より本体1年間
(VALIDITY)　(FULL 1 YEAR AFTER PURCHASE)

お買上げ日　昭和＿＿年＿＿月＿＿日
(PURCHASE)

シャープ株式会社
情報システム事業本部
〒639-11　奈良県大和郡山市美濃庄町492
電話（07435）3−5521番

お客様		
	ご芳名	様
	ご住所	〒
	電話番号	（　　）　−

取扱販売店名・住所・電話番号

㈱ワールドインアオヤマ
東京都豊島区東池袋1-28-6
〒170　電話 03−987−7771

印

シャープ㈱株式会社

本　　社　〒545 大阪市阿倍野区長池町22－22
　　電話（06）621－1221(大代表)
情報システム事業本部　〒639-11 奈良県大和郡山市美濃庄町492
　　電話（07435)3－5521(大代表)
国内情報システム営業本部　〒545 大阪市阿倍野区長池町22－22
　　電話（06）621－1221(大代表)

プログラムや製品、機能についてのお問い合わせは下記へ……

シャープビジネス株式会社

〒545 大阪市阿倍野区長池町22－22　電話（06)621－1221(大代表)

札幌支店	〒063 札幌市西区二十四軒1条7丁目3番17号	電話(011)641－3631
仙台支店	〒983 仙台市6丁目字本新田東2番の1	電話(0222)88－9151
宇都宮支店	〒320 宇都宮市不動前4丁目2番41号	電話(0286)37－3576
東京支店	〒130 東京都墨田区石原2丁目12番3号	電話（03)625－5111
千葉支店	〒280 千葉市南町1丁目5番20号	電話(0472)63－4043
横浜支店	〒235 横浜市磯子区中原1丁目2番23号	電話(045)751－3215
埼玉支店	〒330 埼玉県大宮市宮原町2丁目107番地2号	電話(0486)63－5159
新潟支店	〒950 新潟市上所中1丁目7番21号	電話(0252)83－1795
長野支店	〒380 長野市中御所2丁目11番3号	電話(0262)28－4618
名古屋支店	〒454 名古屋市中川区山王3丁目5番5号	電話(052)332－2631
金沢支店	〒921 石川県石川郡野々市町御経塚町1096の1	電話(0762)49－1240
大阪支店	〒556 大阪市浪速区恵美須西1丁目2番9号	電話（06)643－3021(代表)
京都支店	〒601 京都市南区上鳥羽菅田町48番地	電話(075)661－7335
神戸支店	〒658 神戸市東灘区魚崎北町1丁目6番地18号	電話(078)452－8531
広島支店	〒731-01 広島市安佐南区西原2丁目14番地の3	電話(082)874－4925
高松支店	〒760 香川県高松市木太町1861番地の3	電話(0878)33－4255
福岡支店	〒816 福岡市博多区井相田2丁目12番1号	電話(092)572－2611
沖縄支店	〒900 沖縄県那覇市曙2丁目10番地の1	電話(0988)61－7360(代表)

修理などアフターサービスについてのお問い合わせは下記へ……

シャープお客様ご相談窓口（技術センター）

札幌 技術センター	〒(063) 札幌市西区二十四軒1条7丁目3番17号	電話(011)641－0751
仙台 技術センター	〒(983) 仙台市6丁目字本新田東2番地の1	電話(0222)88－9161
宇都宮 技術センター	〒(320) 宇都宮市不動前4丁目2番41号	電話(0286)34－0256
新潟 技術センター	〒(950) 新潟市上所中1丁目7番21号	電話(0252)84－6023
東京第1技術センター	〒(130) 東京都墨田区石原2丁目12番3号	電話（03)624－7475
東京第2技術センター	〒(164) 東京都中野区南台3丁目45番13号	電話（03)382－9161
東京第3技術センター	〒(143) 東京都大田区南馬込1丁目5番15号	電話（03)777－8851
大宮 技術センター	〒(330) 埼玉県大宮市宮原町2丁目107番2号	電話(0486)66－7220
横浜 技術センター	〒(235) 横浜市磯子区中原1丁目2番23号	電話(045)753－9583
名古屋 技術センター	〒(454) 名古屋市中川区山王3丁目5番5号	電話(052)332－2671
金沢 技術センター	〒(921) 石川県石川郡野々市町字御経塚町1096の1	電話(0762)49－9033
大阪第1技術センター	〒(556) 大阪市浪速区恵美須西1丁目2番9号	電話（06)644－1191
大阪第2技術センター	〒(567) 大阪府茨木市鮎川5丁目15番3号	電話(0726)34－4683
神戸 技術センター	〒(658) 神戸市東灘区魚崎北町1丁目6番18号	電話(078)452－1762
京都 技術センター	〒(601) 京都市南区上鳥羽菅田町48番地	電話(075)681－9551
高松 技術センター	〒(760) 高松市高松町3009番地の4	電話(0878)43－3711
岡山 技術センター	〒(700) 岡山市米倉字東ノ町66の2	電話(0862)43－5660
広島 技術センター	〒(731-01) 広島市安佐南区西原2丁目14番地の3	電話(082)874－6100
福岡 技術センター	〒(816) 福岡市博多区井相田2丁目12番地1号	電話(092)572－2617
鹿児島技術センター	〒(890) 鹿児島市鴨池新町12番1号	電話(0992)59－0628

©1983年

5F3.5O〈TINSJ3975CCZZ〉⑪